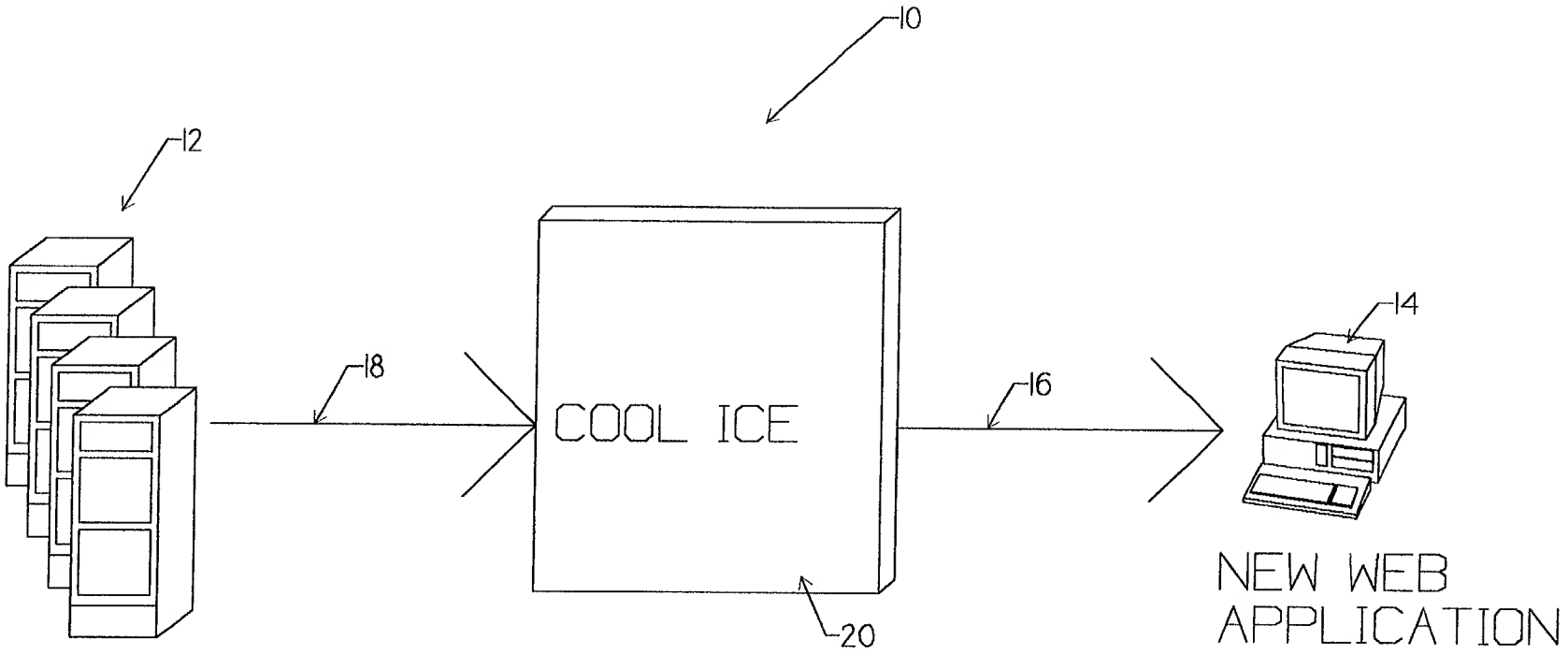
10
12
18
COOL ICE
16
14
20
NEW WEB APPLICATION
EXISTING DATABASES AND APPLICATIONS

FIG. 1

PATENT APPLICATION

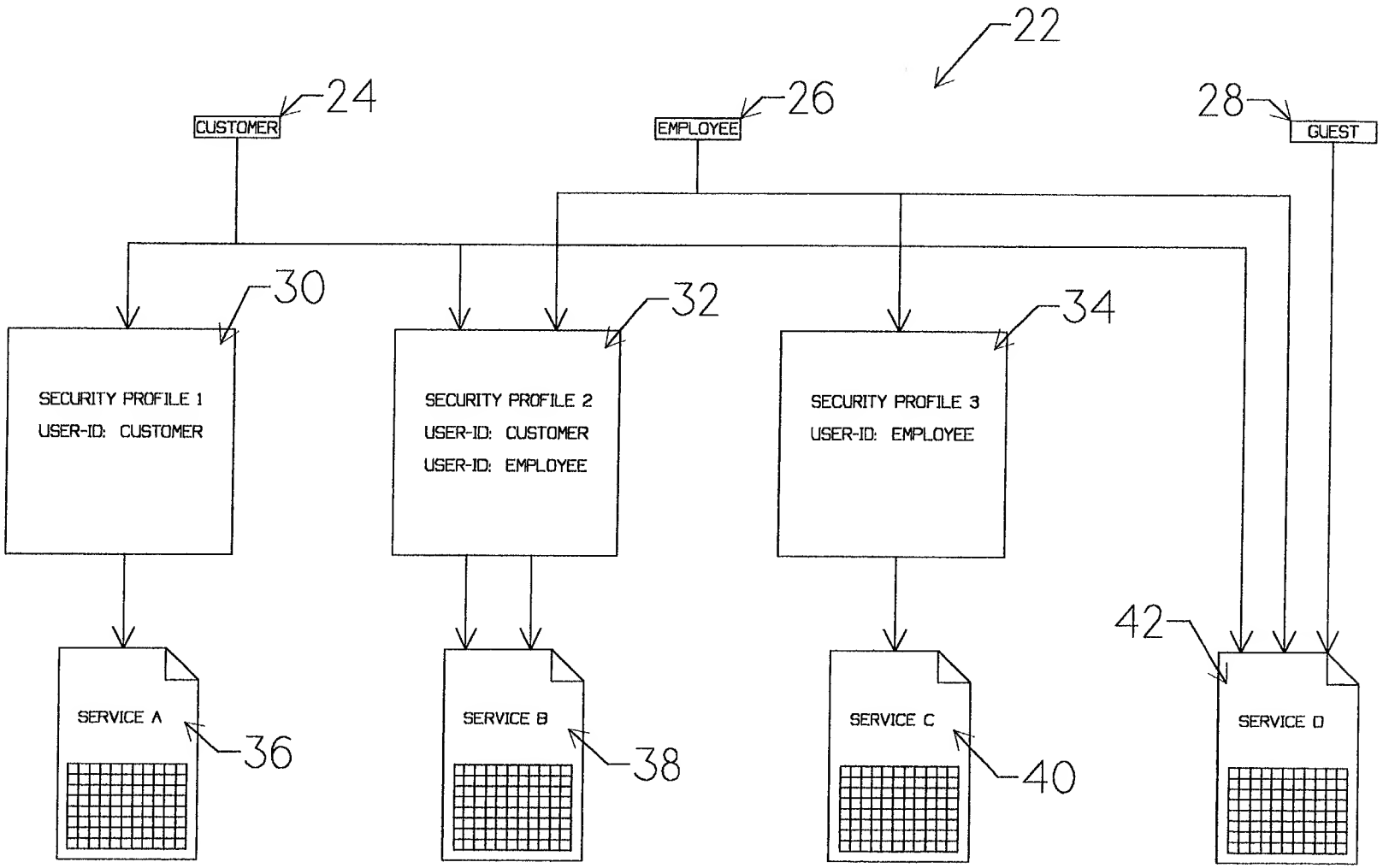

FIG. 2

PATENT APPLICATION

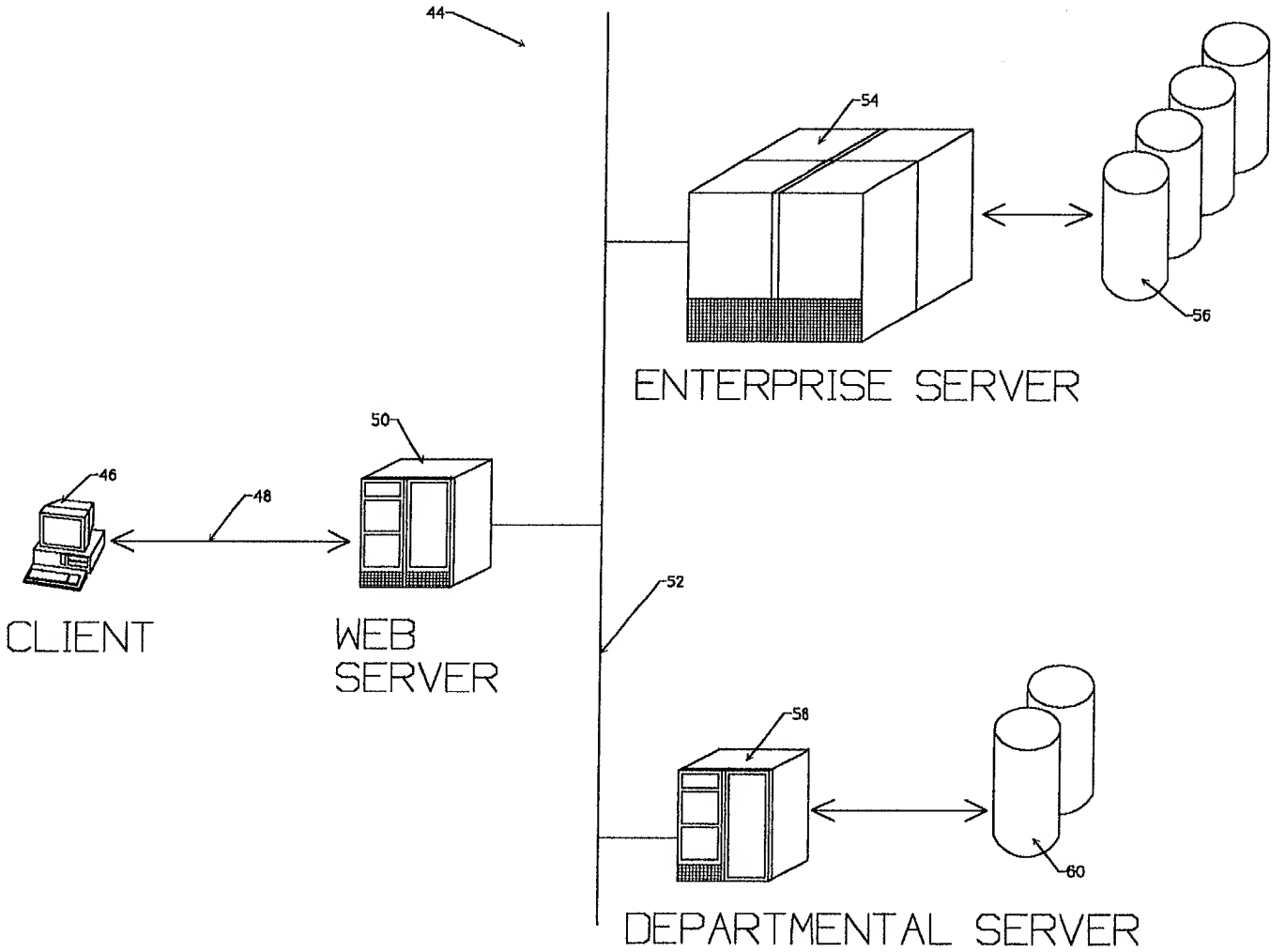

FIG. 3

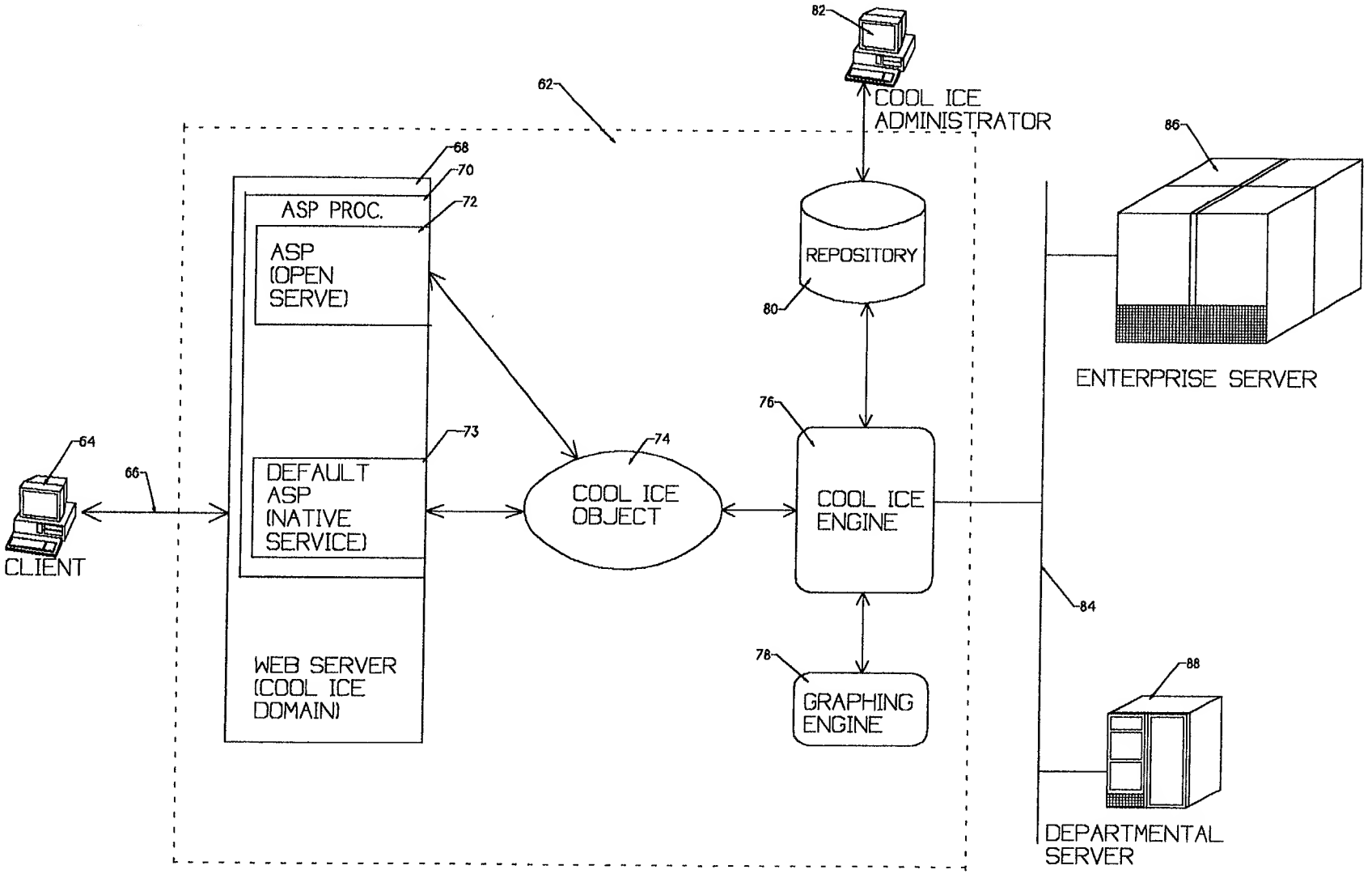
82
COOL ICE
ADMINISTRATOR
62
68
70
72
ASP PROC.
ASP
(OPEN
SERVE)
REPOSITORY
80
86
ENTERPRISE SERVER
76
64
66
73
74
DEFAULT
ASP
(NATIVE
SERVICE)
COOL ICE
OBJECT
COOL ICE
ENGINE
CLIENT
84
88
78
GRAPHING
ENGINE
WEB SERVER
(COOL ICE
DOMAIN)
DEPARTMENTAL
SERVER

FIG. 4

PAGE 1 OF 1

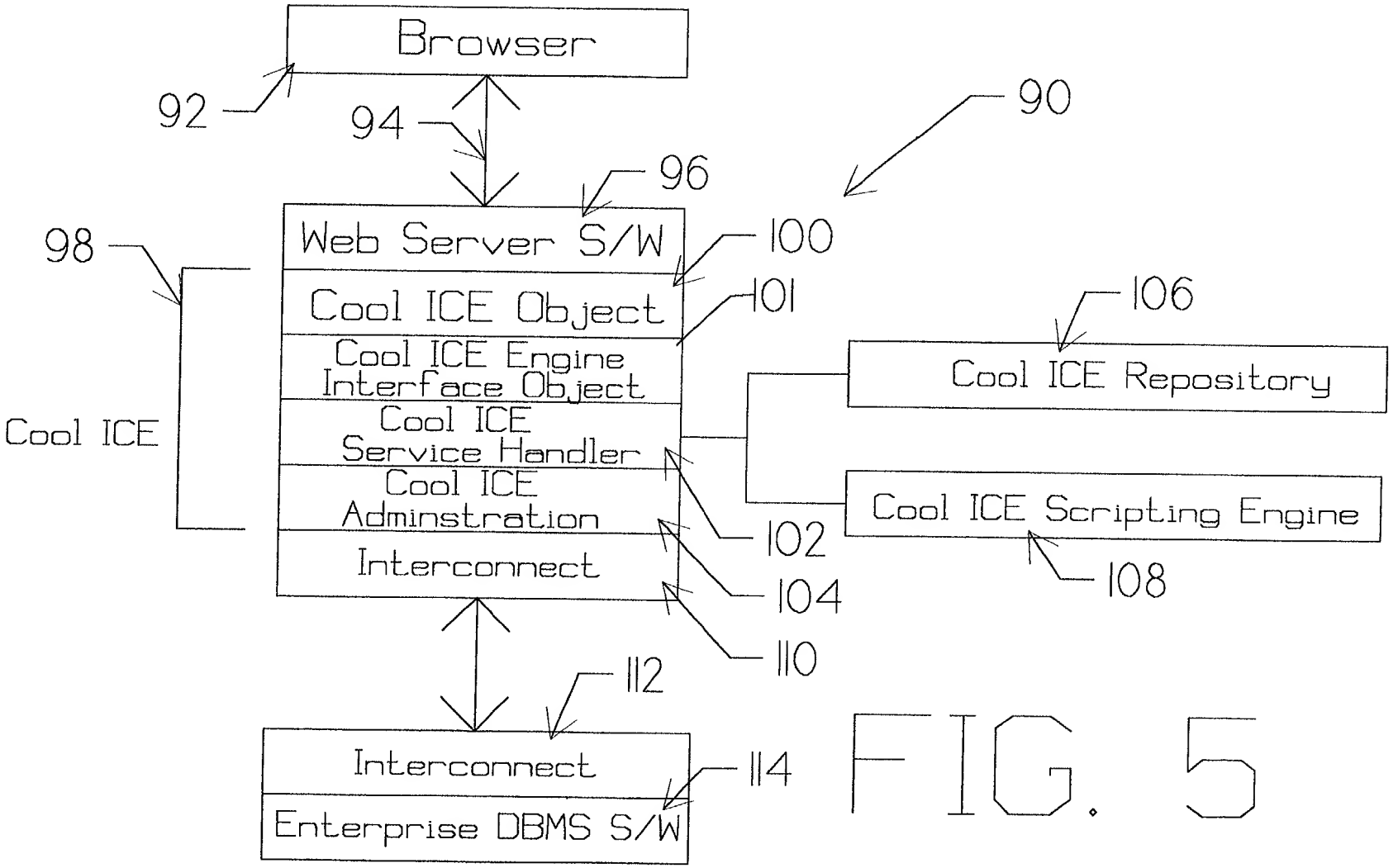

FIG. 5

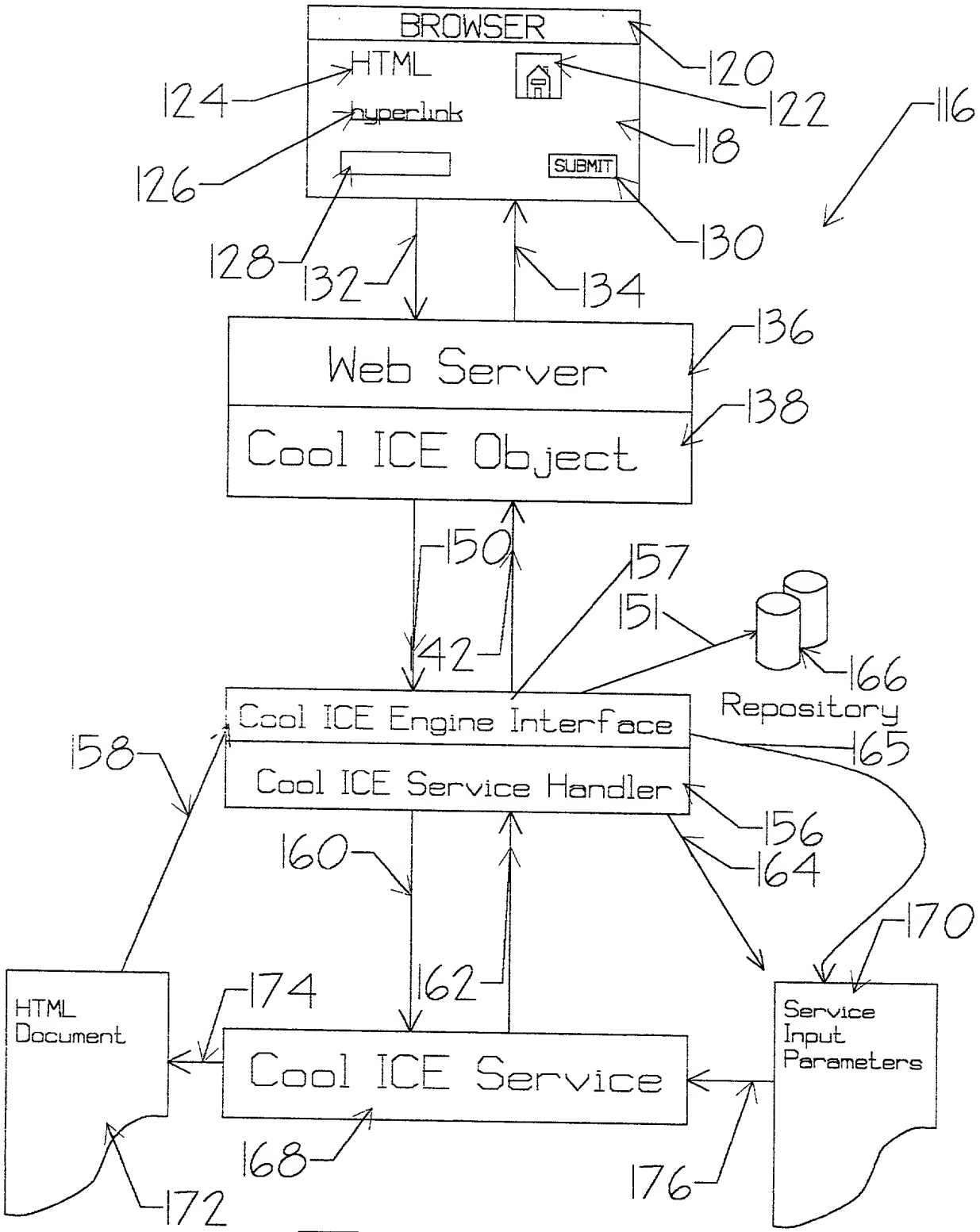
BROWSER
HTML
hyperlink
SUBMIT
120
122
124
126
116
118
128
130
132
134
Web Server
Cool ICE Object
136
138
150
157
151
42
166
Repository
Cool ICE Engine Interface
Cool ICE Service Handler
165
158
156
160
164
170
162
174
HTML Document
Service Input Parameters
Cool ICE Service
168
176
172

FIG. 6

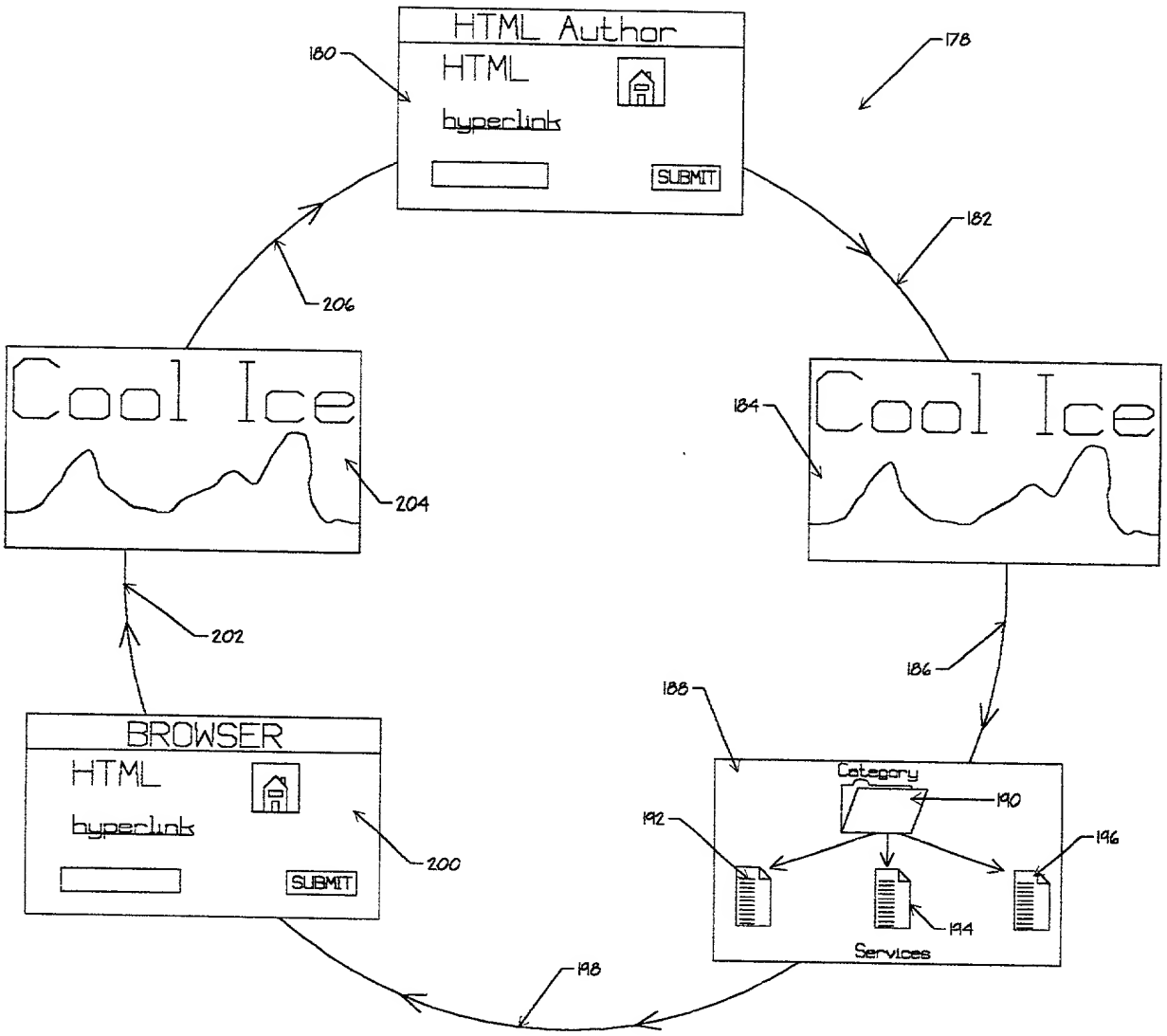
HTML Author
HTML
hyperlink
SUBMIT
180
178
182
206
Cool Ice
204
184
Cool Ice
202
186
188
BROWSER
HTML
hyperlink
SUBMIT
200
Category
190
192
196
194
Services
198

FIG. 7

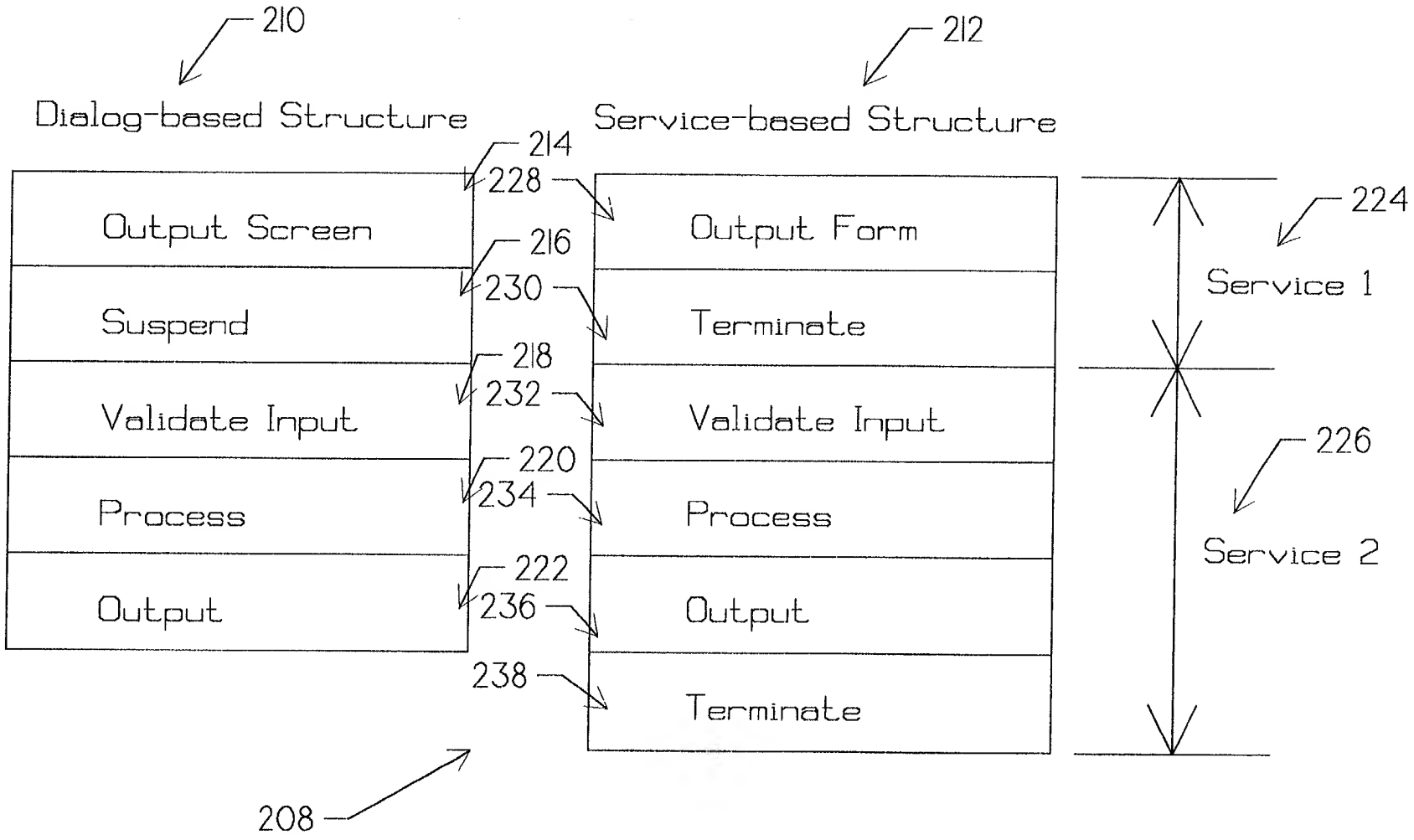
210
Dialog-based Structure
212
Service-based Structure
214
228
216
230
218
232
220
234
222
236
238
Output Screen
Suspend
Validate Input
Process
Output
Output Form
Terminate
Validate Input
Process
Output
Terminate
224
Service 1
226
Service 2
208

FIG. 8

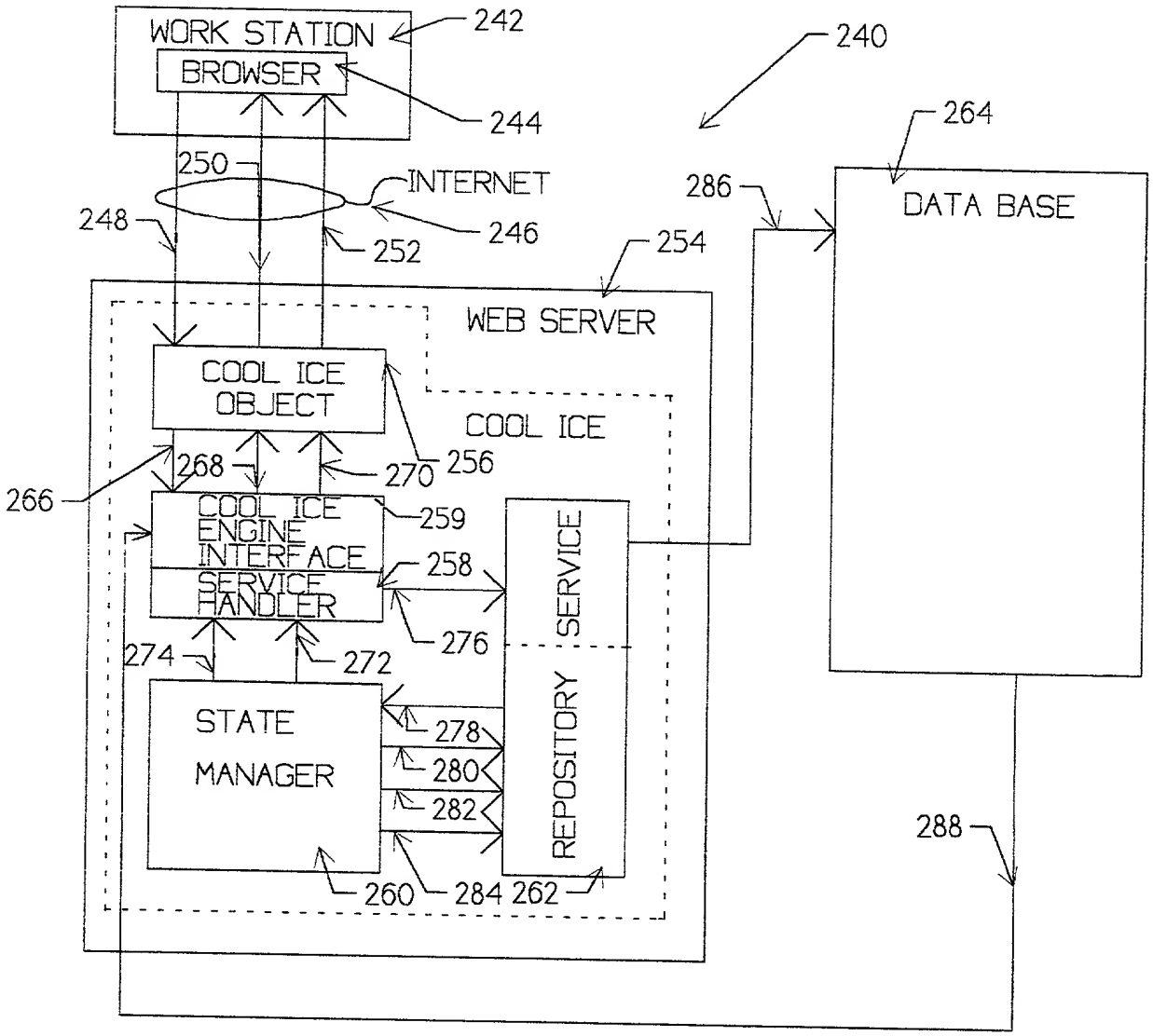
WORK STATION
BROWSER
242
244
240
264
250
INTERNET
286
DATA BASE
248
246
252
254
WEB SERVER
COOL ICE OBJECT
COOL ICE
268
270
256
266
COOL ICE ENGINE INTERFACE
259
SERVICE HANDLER
258
276
SERVICE
274
272
STATE MANAGER
278
280
282
REPOSITORY
288
260
284
262

FIG. 9

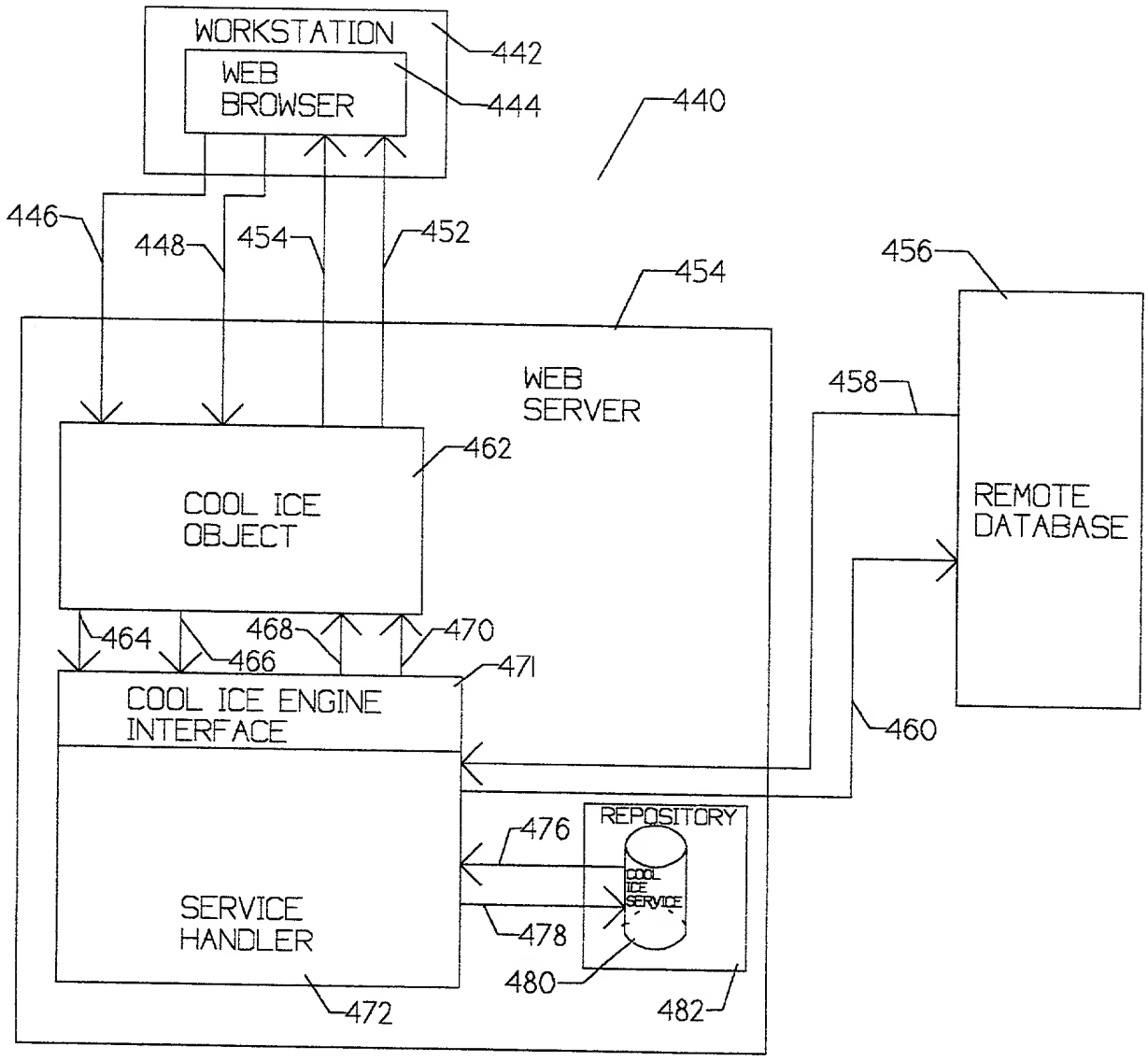
WORKSTATION
442
WEB
BROWSER
444
440
446
448
454
452
456
454
WEB
SERVER
458
462
COOL ICE
OBJECT
REMOTE
DATABASE
464
468
470
466
471
COOL ICE ENGINE
INTERFACE
460
REPOSITORY
476
COOL
ICE
SERVICE
SERVICE
HANDLER
478
480
472
482

FIG. 10

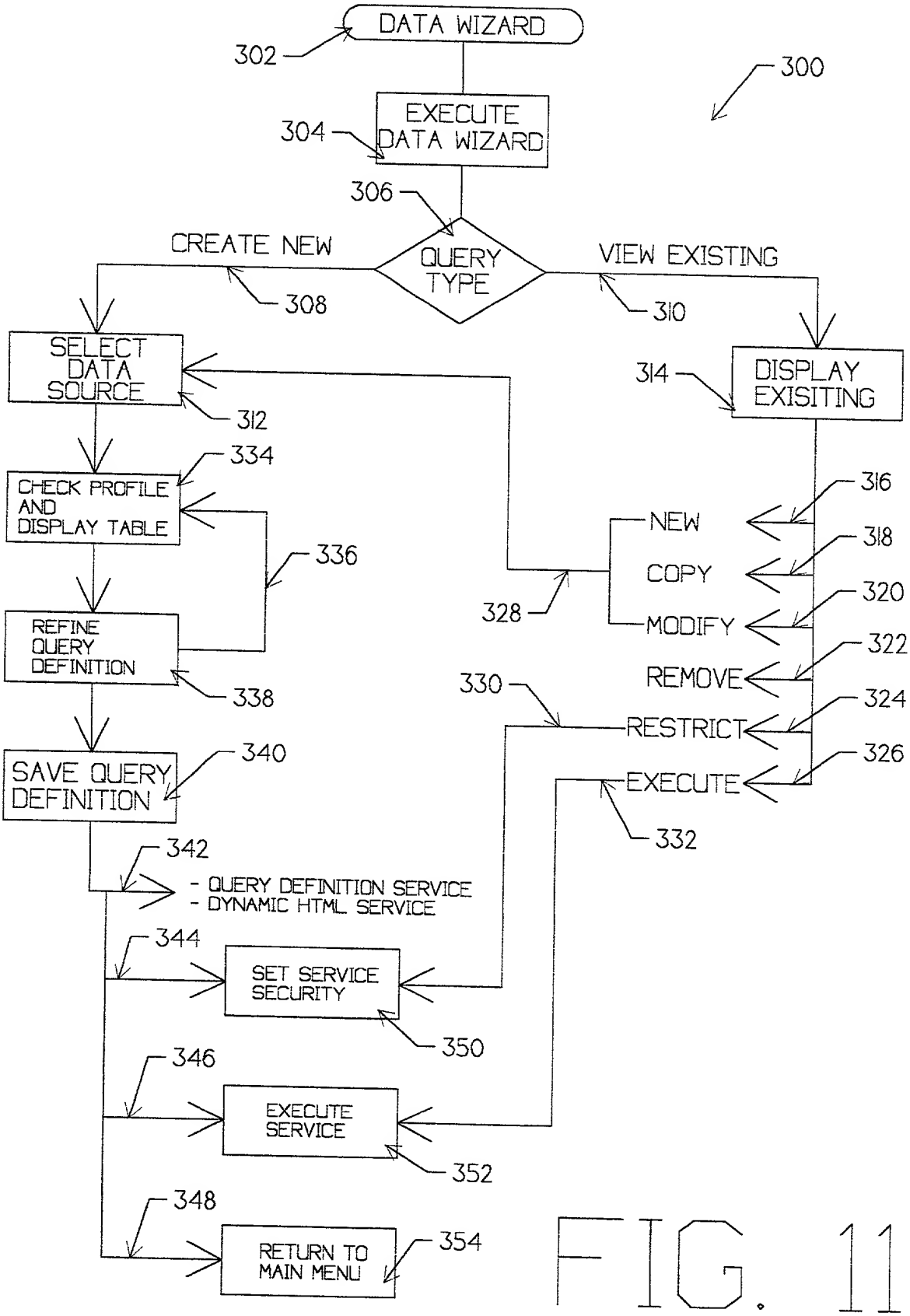
DATA WIZARD
302
300
EXECUTE DATA WIZARD
304
306
QUERY TYPE
CREATE NEW
308
VIEW EXISTING
310
SELECT DATA SOURCE
312
314
DISPLAY EXISITING
334
CHECK PROFILE AND DISPLAY TABLE
336
REFINE QUERY DEFINITION
338
SAVE QUERY DEFINITION
340
328
NEW
COPY
MODIFY
REMOVE
RESTRICT
EXECUTE
316
318
320
322
324
326
330
332
342
- QUERY DEFINITION SERVICE
- DYNAMIC HTML SERVICE
344
SET SERVICE SECURITY
350
346
EXECUTE SERVICE
352
348
RETURN TO MAIN MENU
354

FIG. 11

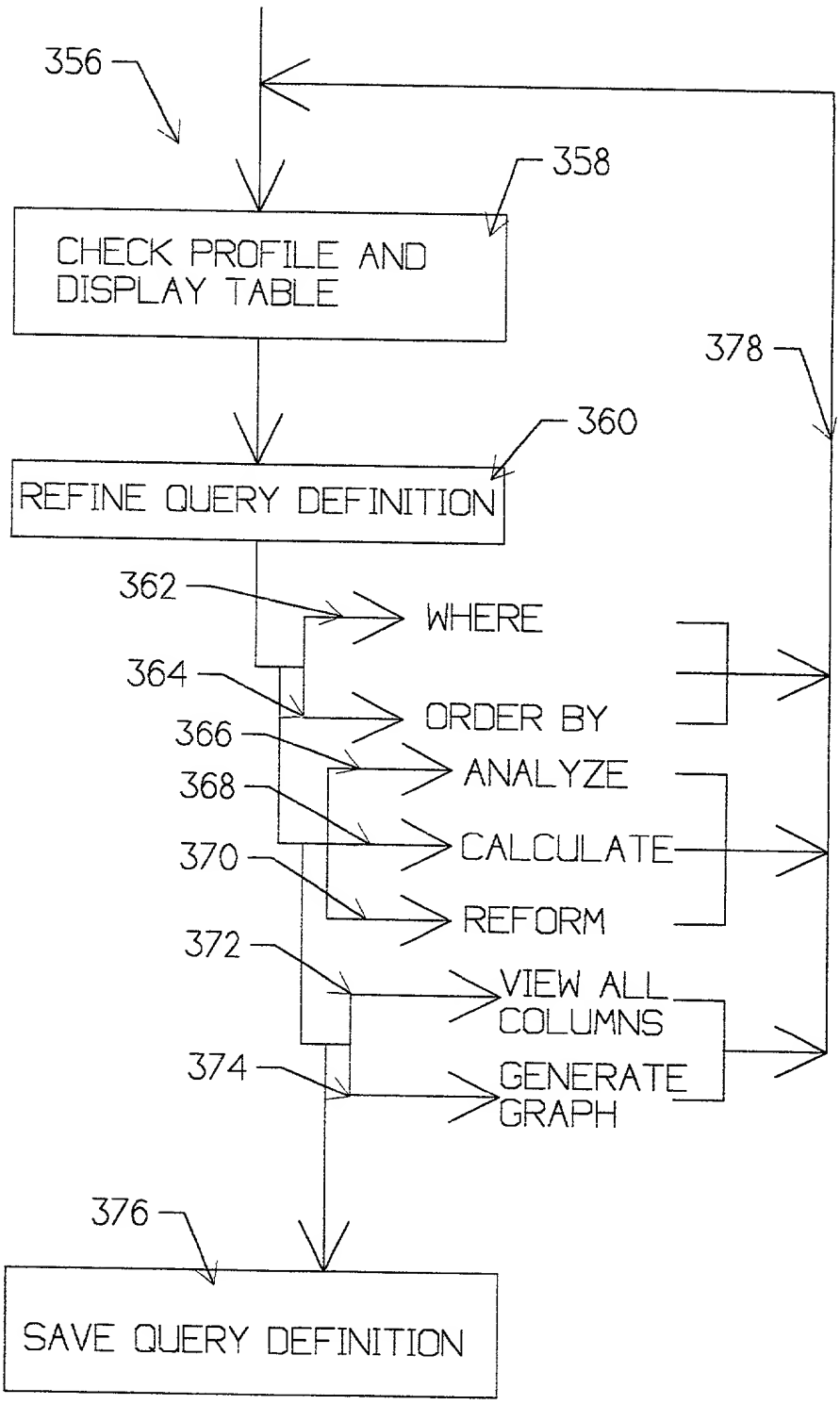
356
358
CHECK PROFILE AND DISPLAY TABLE
378
360
REFINE QUERY DEFINITION
362
WHERE
364
ORDER BY
366
ANALYZE
368
CALCULATE
370
REFORM
372
VIEW ALL COLUMNS
374
GENERATE GRAPH
376
SAVE QUERY DEFINITION

FIG. 12

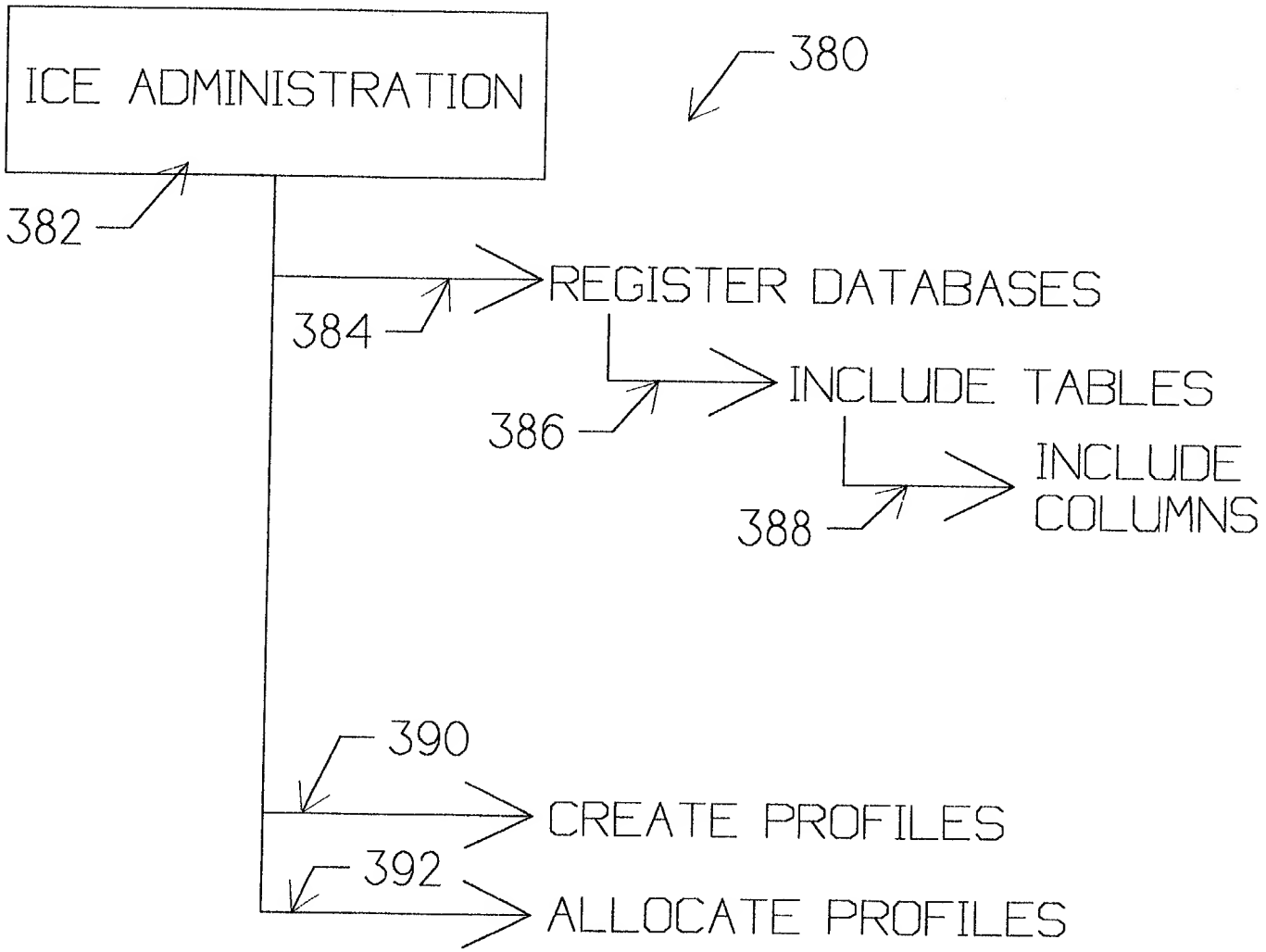
ICE ADMINISTRATION
380
382
REGISTER DATABASES
384
INCLUDE TABLES
386
INCLUDE COLUMNS
388
390
CREATE PROFILES
392
ALLOCATE PROFILES

FIG. 13

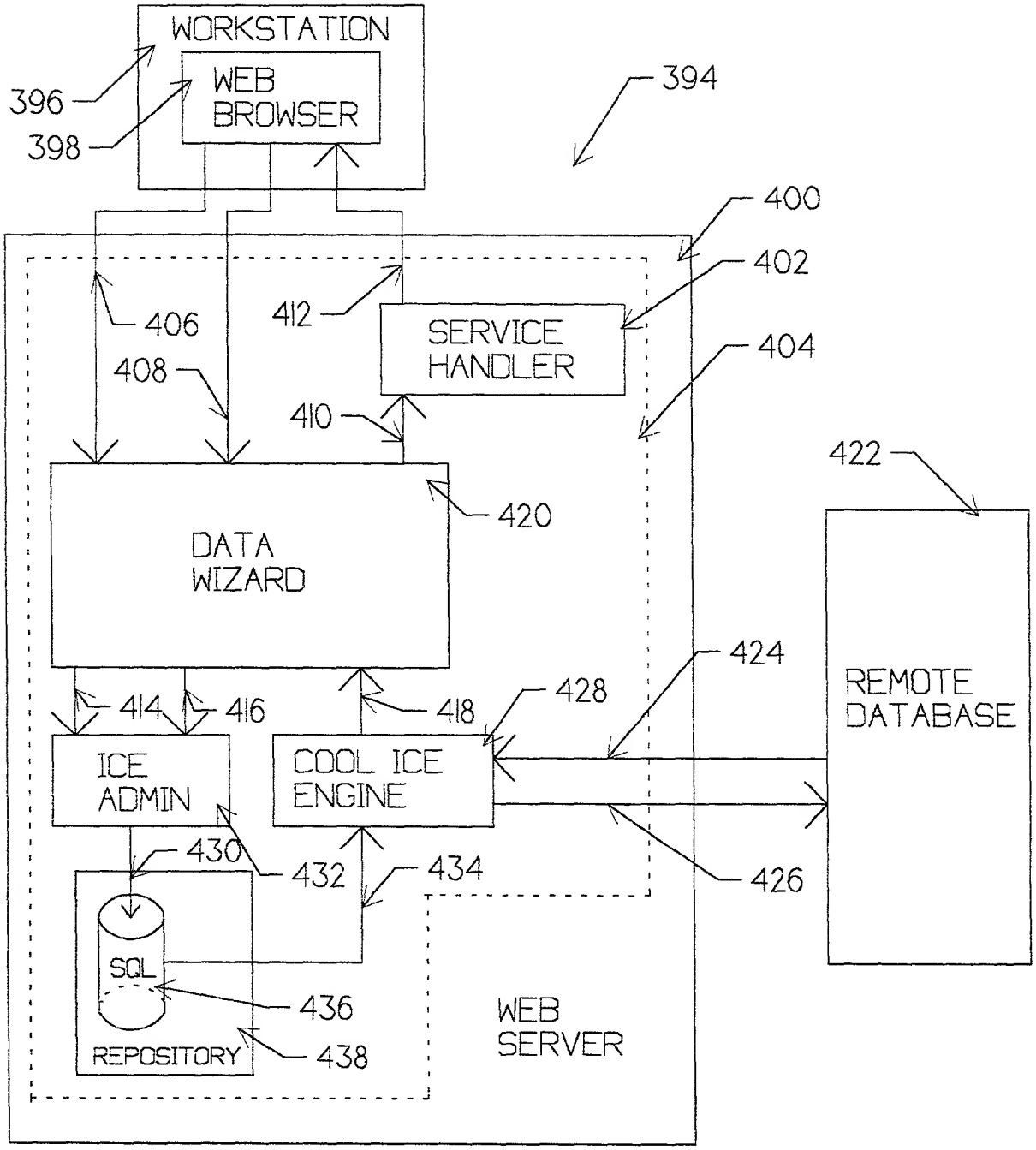
WORKSTATION
WEB BROWSER
396
398
394
400
402
404
406
408
410
412
SERVICE HANDLER
DATA WIZARD
420
422
REMOTE DATABASE
424
426
414
416
418
428
ICE ADMIN
COOL ICE ENGINE
430
432
434
SQL
436
REPOSITORY
438
WEB SERVER

FIG. 14

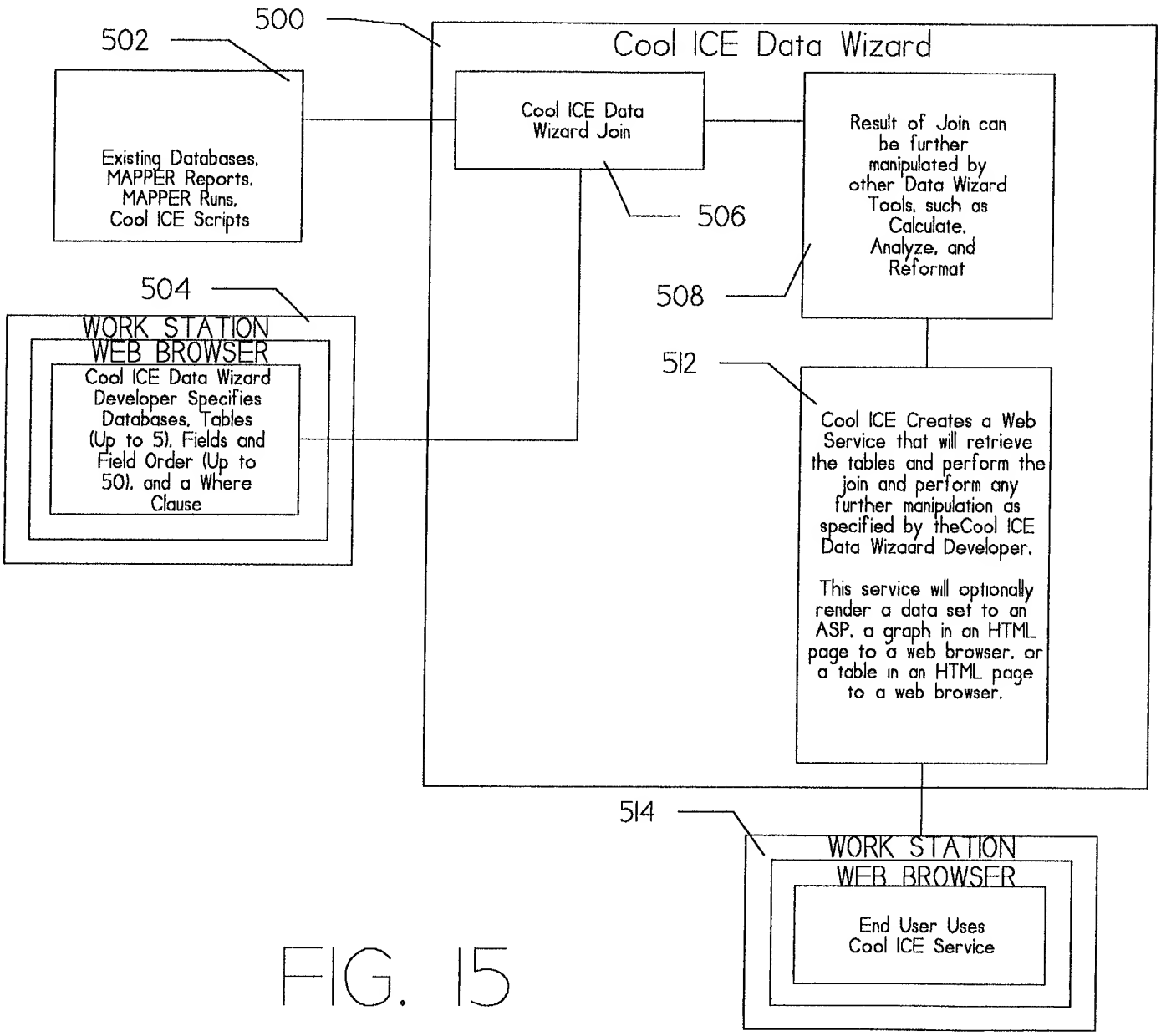
500
Cool ICE Data Wizard
502
Existing Databases, MAPPER Reports, MAPPER Runs, Cool ICE Scripts
Cool ICE Data Wizard Join
506
Result of Join can be further manipulated by other Data Wizard Tools, such as Calculate, Analyze, and Reformat
508
504
WORK STATION WEB BROWSER
Cool ICE Data Wizard Developer Specifies Databases, Tables (Up to 5), Fields and Field Order (Up to 50), and a Where Clause
512
Cool ICE Creates a Web Service that will retrieve the tables and perform the join and perform any further manipulation as specified by theCool ICE Data Wizaard Developer.
This service will optionally render a data set to an ASP, a graph in an HTML page to a web browser, or a table in an HTML page to a web browser.
514
WORK STATION WEB BROWSER
End User Uses Cool ICE Service

FIG. 15

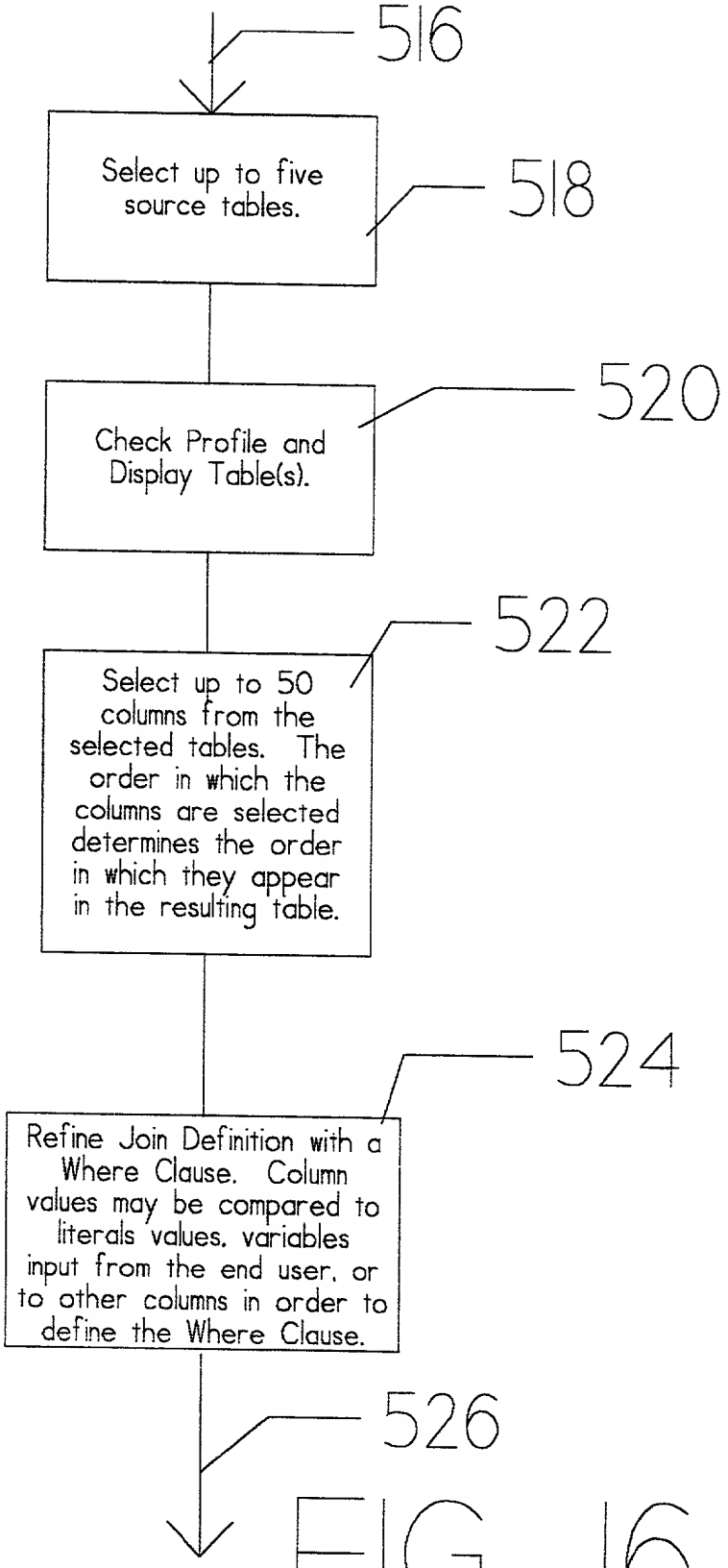
516
Select up to five source tables.
518
Check Profile and Display Table(s).
520
Select up to 50 columns from the selected tables. The order in which the columns are selected determines the order in which they appear in the resulting table.
522
Refine Join Definition with a Where Clause. Column values may be compared to literals values, variables input from the end user, or to other columns in order to define the Where Clause.
524
526

FIG. 16

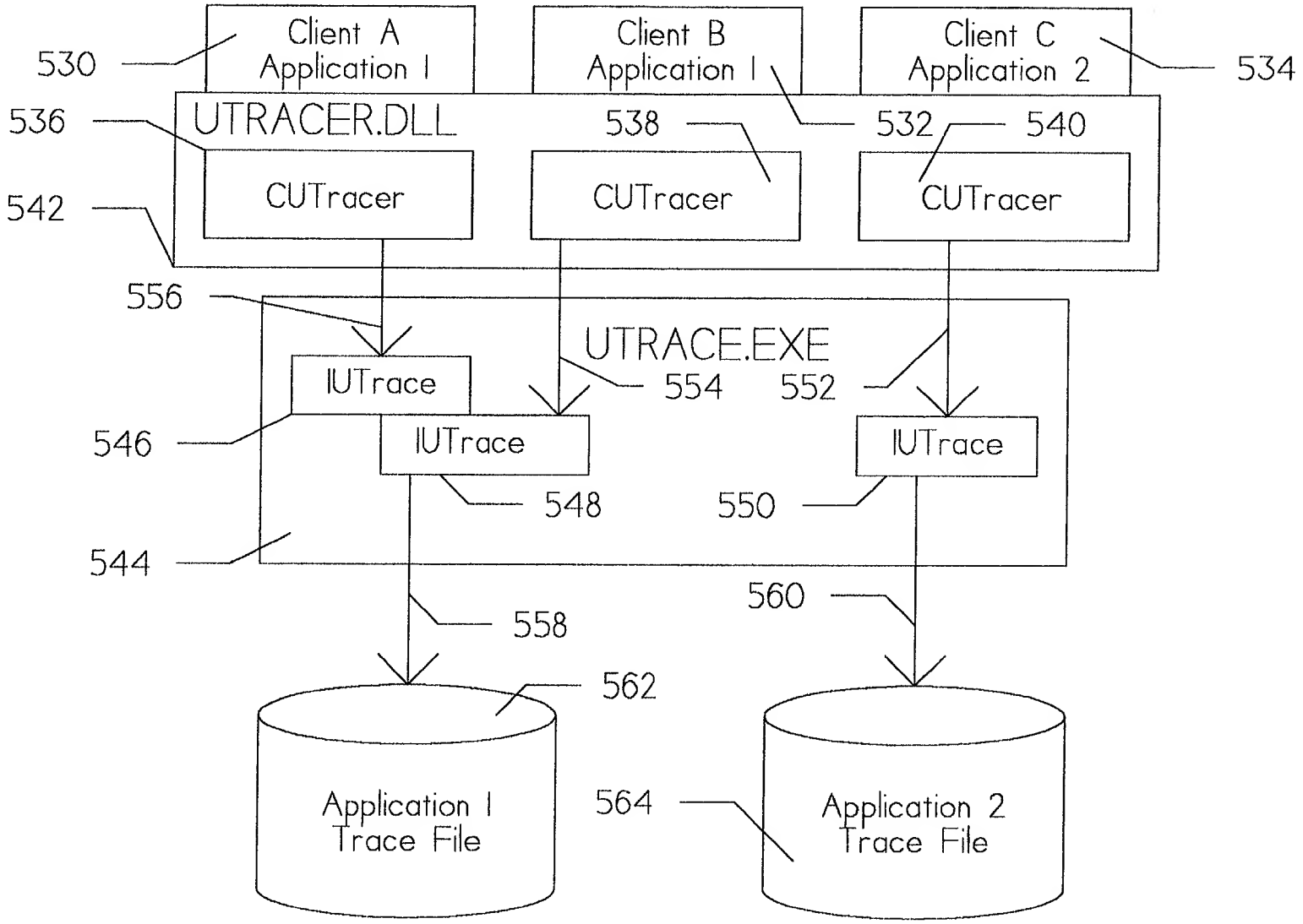
Client A
Application 1
Client B
Application 1
Client C
Application 2
530
534
536
UTRACER.DLL
538
532
540
542
CUTracer
CUTracer
CUTracer
556
UTRACE.EXE
IUTrace
554
552
546
IUTrace
IUTrace
548
550
544
560
558
562
Application 1
Trace File
564
Application 2
Trace File

FIG. 17

HKLM\SOFTWARE\UNISYS\UTRACE

APPNAME -- Subkey matching the name of the application

565

- ENABLETRACE -- DWORD value indication a Boolean (1=trace,, 0=no trace).
- MAXENTRIES -- DWORD value indicating the maximum number of entries in the trace file. If missing, then no upper limit is imposed.
- FORMATFLAGS -- DWORD value indicating message formatting flags.
- POLICYFLAGS -- DWORD value indicating trace policy flags set for the application.
- TRACELEVEL -- DWORD value indicating a trace level. 0 = no trace.
- TRACEPATH -- String value indicating the directory path for the trace files.
- REGTRACE -- A string value containing the HKLM\Software sub key to trace for this application. Blank or missing for no registry trace.
- COMPONENT -- A subkey for a specific component of the application.
  - POLICYFLAGS -- Value indicating component level trace policy flags
  - TRACELEVEL -- DWORD Value indicating a trace level. 0 = no trace.
  - VERSIONTRACE-- String value specifying executable component name for version information tracing.

FIG. 18

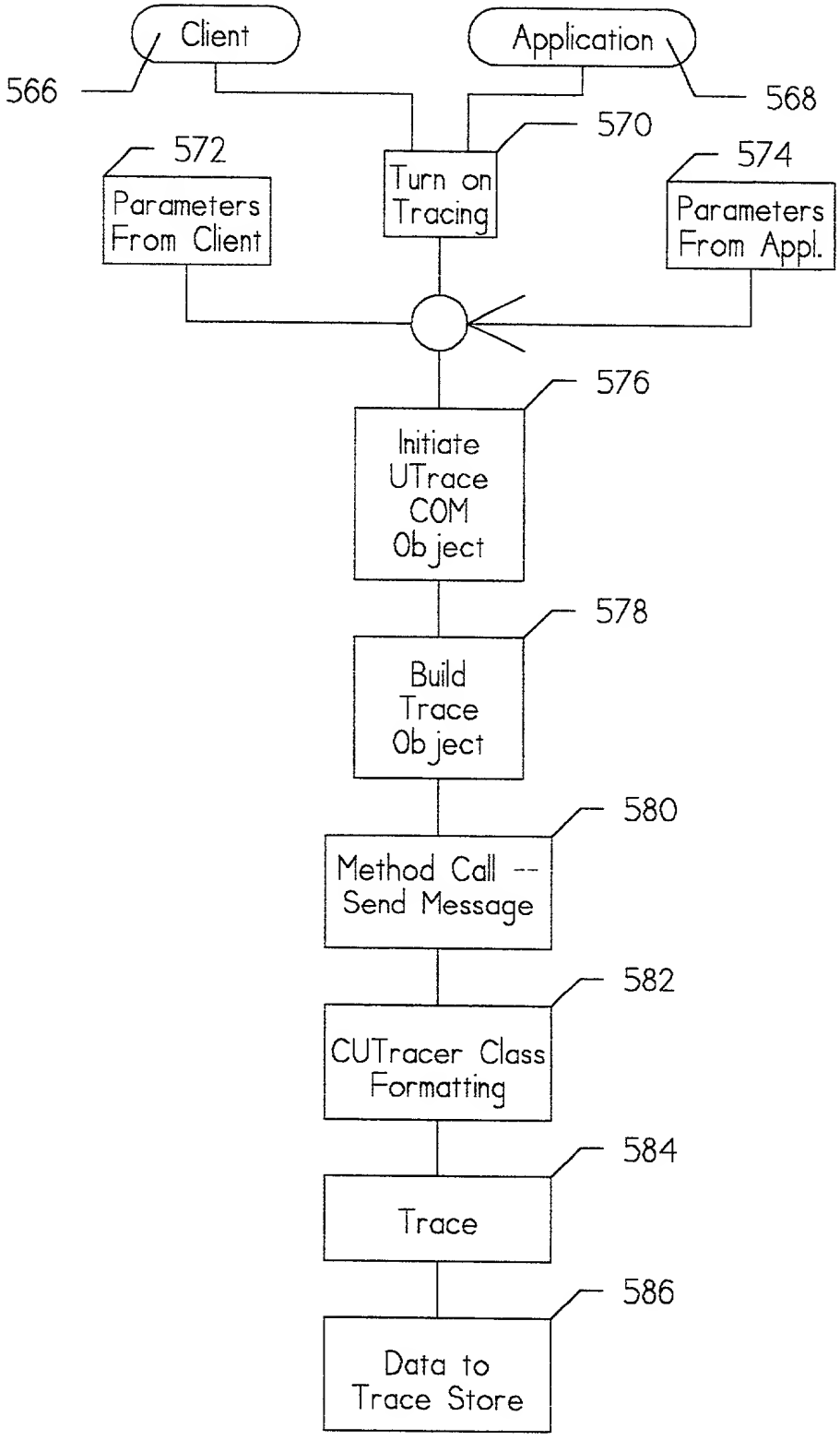
Client
Application
566
568
570
572
574
Turn on Tracing
Parameters From Client
Parameters From Appl.
576
Initiate UTrace COM Object
578
Build Trace Object
580
Method Call -- Send Message
582
CUTracer Class Formatting
584
Trace
586
Data to Trace Store

FIG. 19

```
#define CI_TRACE_VERSION          0x1
#define CI_TRACE_ERROR            0x2
#define CI_TRACE_INTERFACE        0x4
#define CI_TRACE_FLOW             0x8
#define CI_TRACE_DETAIL           0x10
```

FIG. 20A

```
if (m_trace. Active(C)_TRACE_DETAIL))
m_trace<<"MY Detailed Trace information"<<Localvariable<<traceit
```

FIG. 20B

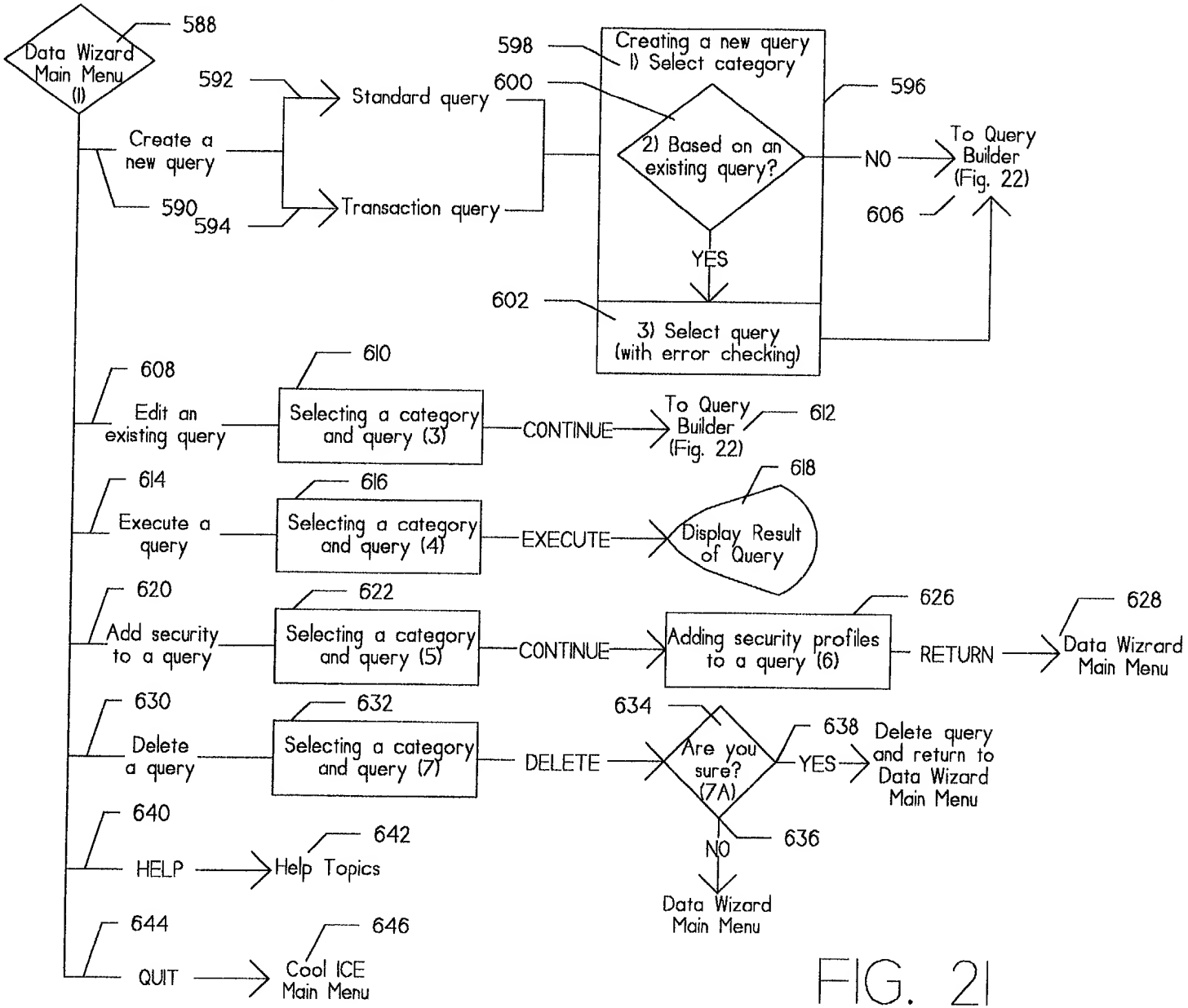
588
Data Wizard Main Menu (1)
592
Create a new query
590
Standard query
594
Transaction query
598
600
Creating a new query 1) Select category
2) Based on an existing query?
596
NO
To Query Builder (Fig. 22)
606
YES
602
3) Select query (with error checking)
608
Edit an existing query
610
Selecting a category and query (3)
CONTINUE
To Query Builder (Fig. 22)
612
614
Execute a query
616
Selecting a category and query (4)
EXECUTE
618
Display Result of Query
620
Add security to a query
622
Selecting a category and query (5)
CONTINUE
626
Adding security profiles to a query (6)
RETURN
628
Data Wizrard Main Menu
630
Delete a query
632
Selecting a category and query (7)
DELETE
634
Are you sure? (7A)
638
YES
Delete query and return to Data Wizard Main Menu
636
NO
Data Wizard Main Menu
640
HELP
642
Help Topics
644
QUIT
646
Cool ICE Main Menu

FIG. 21

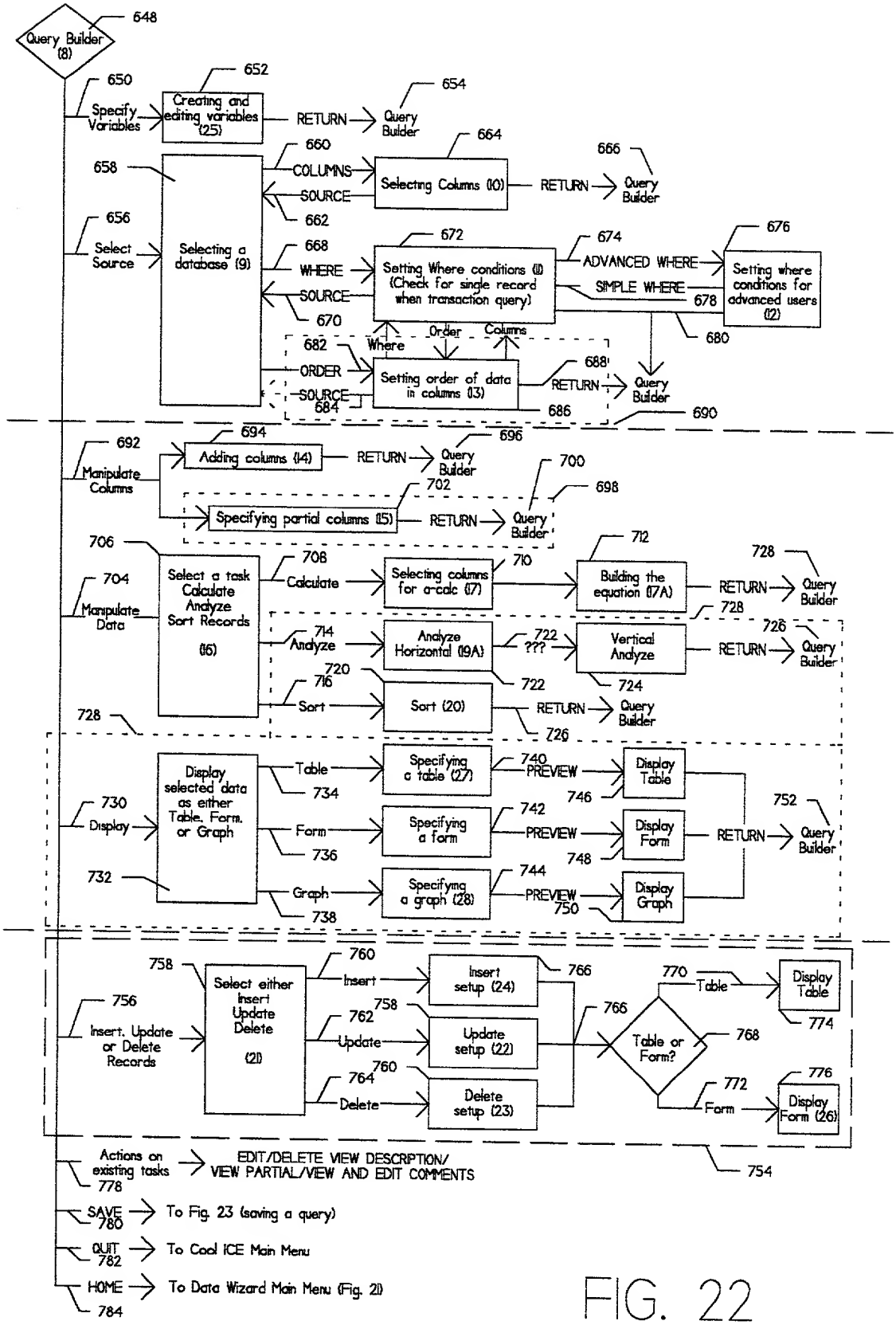
Query Builder (8)
648
650
Specify Variables
652
Creating and editing variables (25)
RETURN
654
Query Builder
658
656
Select Source
Selecting a database (9)
660
COLUMNS
SOURCE
662
664
Selecting Columns (10)
RETURN
666
Query Builder
668
WHERE
SOURCE
670
672
Setting Where conditions (11) (Check for single record when transaction query)
674
ADVANCED WHERE
SIMPLE WHERE
678
676
Setting where conditions for advanced users (12)
680
682
ORDER
SOURCE
684
Where
Order
Columns
Setting order of data in columns (13)
688
RETURN
Query Builder
686
690
692
Manipulate Columns
694
Adding columns (14)
RETURN
696
Query Builder
700
698
702
Specifying partial columns (15)
RETURN
Query Builder
706
704
Manipulate Data
Select a task Calculate Analyze Sort Records (16)
708
Calculate
Selecting columns for a-calc (17)
710
712
Building the equation (17A)
RETURN
728
Query Builder
714
Analyze
Analyze Horizontal (19A)
722
???
Vertical Analyze
RETURN
726
Query Builder
728
720
716
Sort
Sort (20)
RETURN
Query Builder
726
728
730
Display
Display selected data as either Table, Form, or Graph
732
Table
734
Specifying a table (27)
740
PREVIEW
Display Table
746
Form
736
Specifying a form
742
PREVIEW
Display Form
748
RETURN
752
Query Builder
Graph
738
Specifying a graph (28)
744
PREVIEW
750
Display Graph
758
756
Insert, Update or Delete Records
Select either Insert Update Delete (21)
760
Insert
Insert setup (24)
766
762
Update
Update setup (22)
766
764
Delete
Delete setup (23)
760
Table or Form?
770
Table
Display Table
774
768
772
Form
Display Form (26)
776
754
Actions on existing tasks
778
EDIT/DELETE VIEW DESCRIPTION/ VIEW PARTIAL/VIEW AND EDIT COMMENTS
SAVE
780
To Fig. 23 (saving a query)
QUIT
782
To Cool ICE Main Menu
HOME
784
To Data Wizard Main Menu (Fig. 21)

FIG. 22

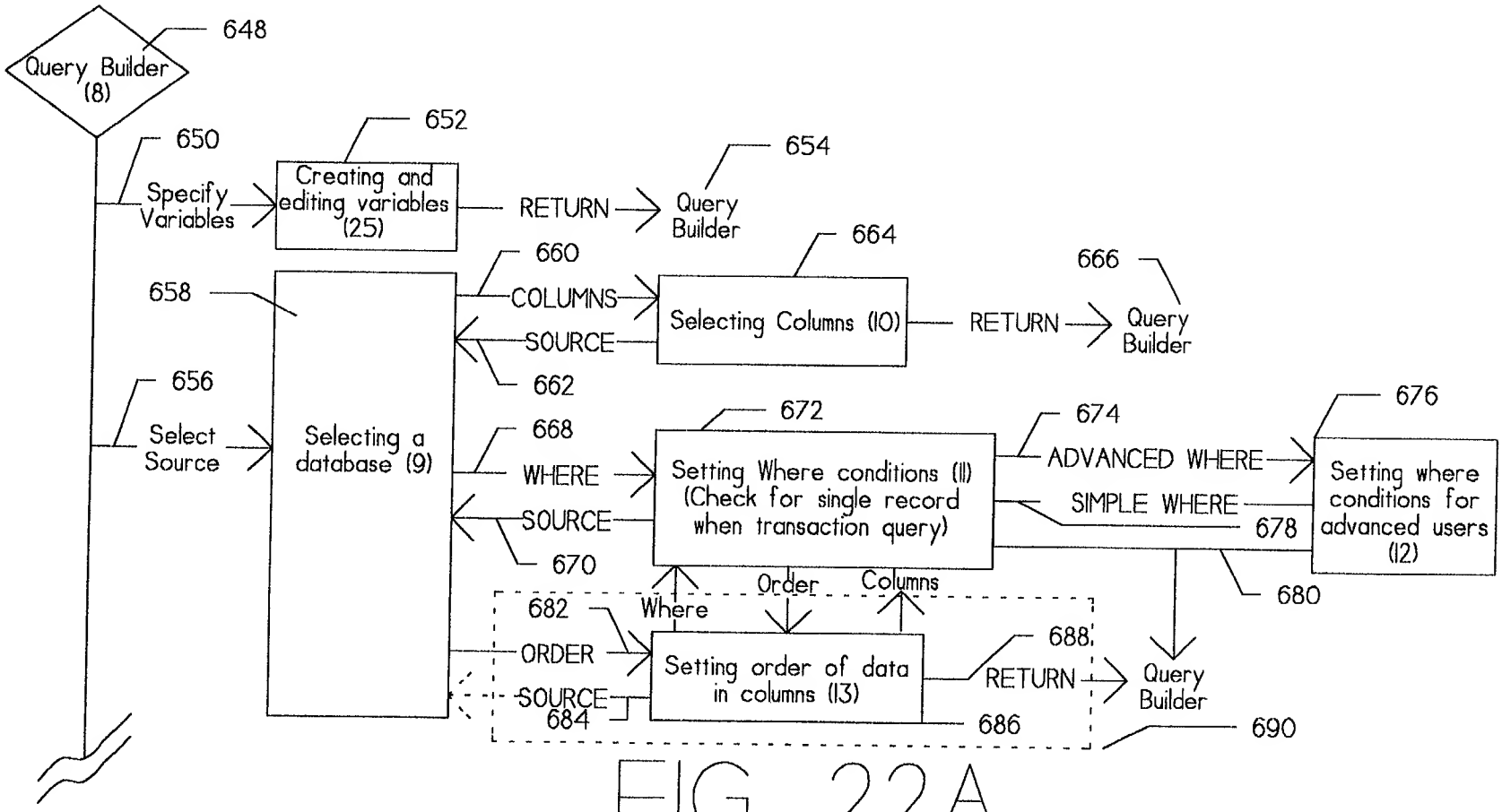
Query Builder (8)
648
650
Specify Variables
652
Creating and editing variables (25)
RETURN
654
Query Builder
656
Select Source
658
Selecting a database (9)
660
COLUMNS
SOURCE
662
664
Selecting Columns (10)
RETURN
666
Query Builder
668
WHERE
SOURCE
670
672
Setting Where conditions (11) (Check for single record when transaction query)
674
ADVANCED WHERE
SIMPLE WHERE
678
676
Setting where conditions for advanced users (12)
680
Where
Order
Columns
682
ORDER
SOURCE
684
Setting order of data in columns (13)
688
RETURN
686
Query Builder
690

FIG. 22A

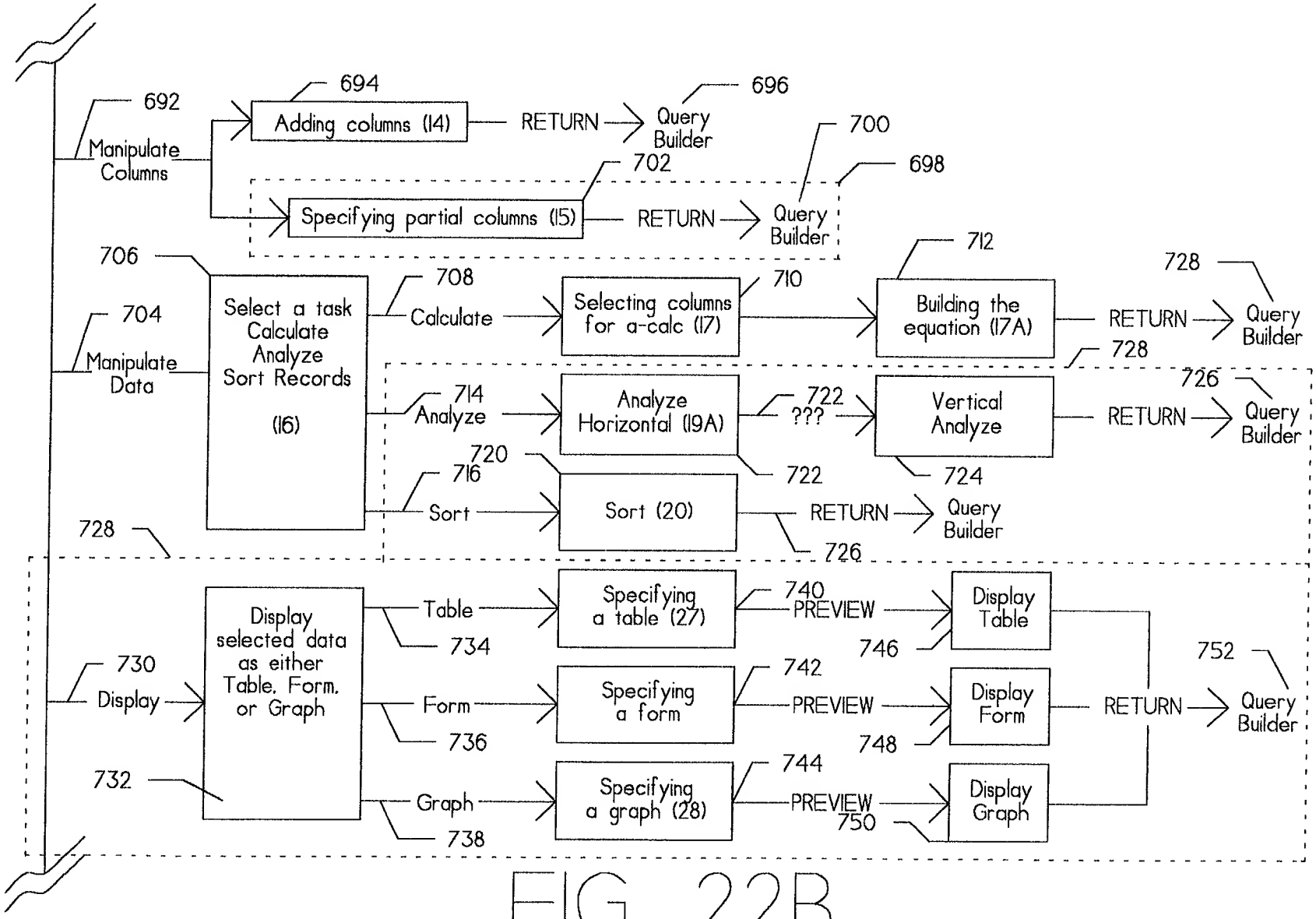
692
Manipulate Columns
694
Adding columns (14)
RETURN
696
Query Builder
700
702
Specifying partial columns (15)
RETURN
698
Query Builder
706
704
Manipulate Data
Select a task
Calculate
Analyze
Sort Records
(16)
708
Calculate
Selecting columns for a-calc (17)
710
712
Building the equation (17A)
RETURN
728
Query Builder
728
714
Analyze
Analyze Horizontal (19A)
722
???
Vertical Analyze
RETURN
726
Query Builder
720
716
Sort
Sort (20)
722
724
RETURN
Query Builder
726
728
730
Display
Display selected data as either Table, Form, or Graph
732
Table
734
Specifying a table (27)
740
PREVIEW
Display Table
746
Form
736
Specifying a form
742
PREVIEW
Display Form
748
RETURN
752
Query Builder
Graph
738
Specifying a graph (28)
744
PREVIEW
750
Display Graph

FIG. 22B

PROOF SUBMITTED

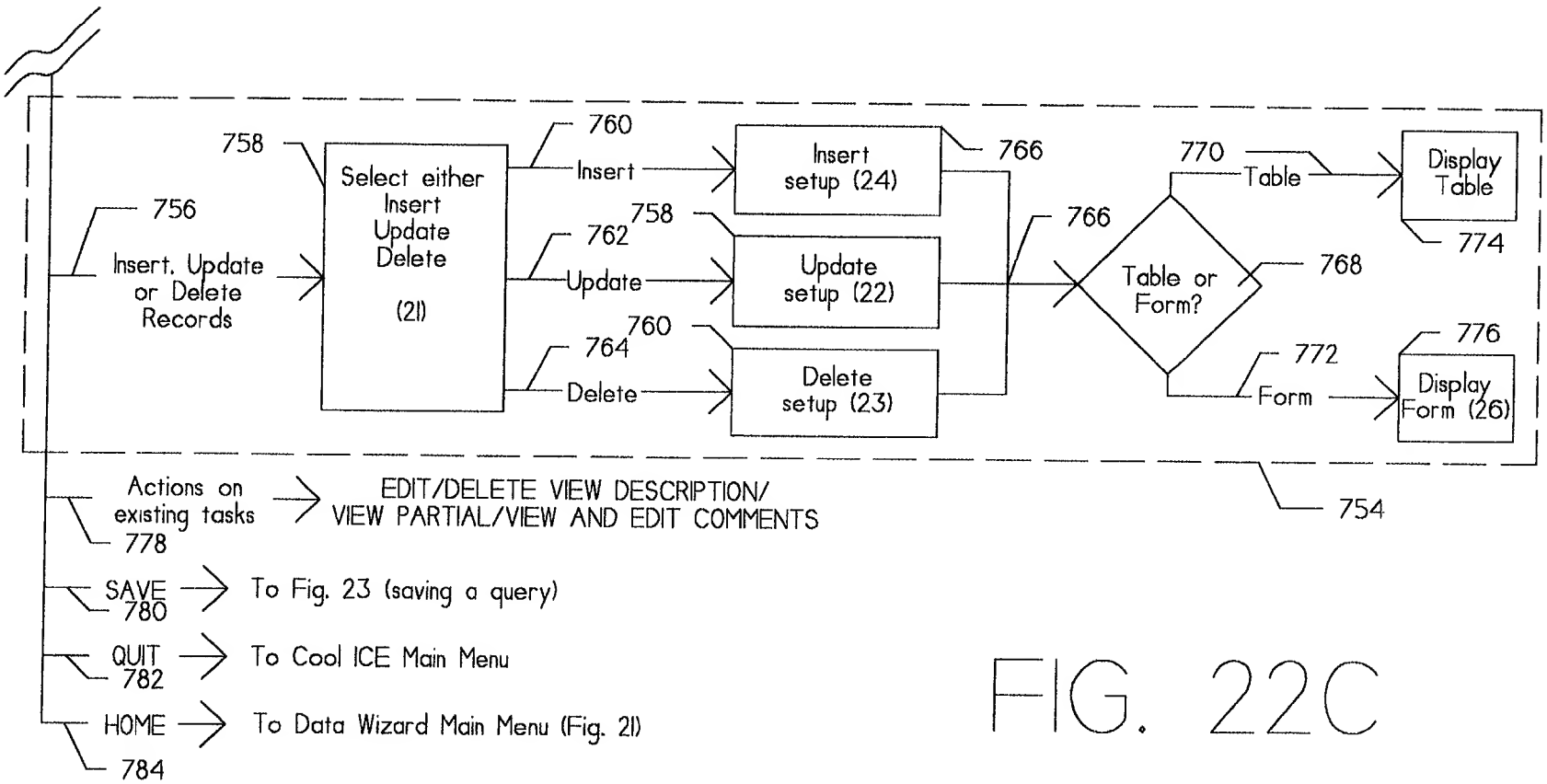

FIG. 22C

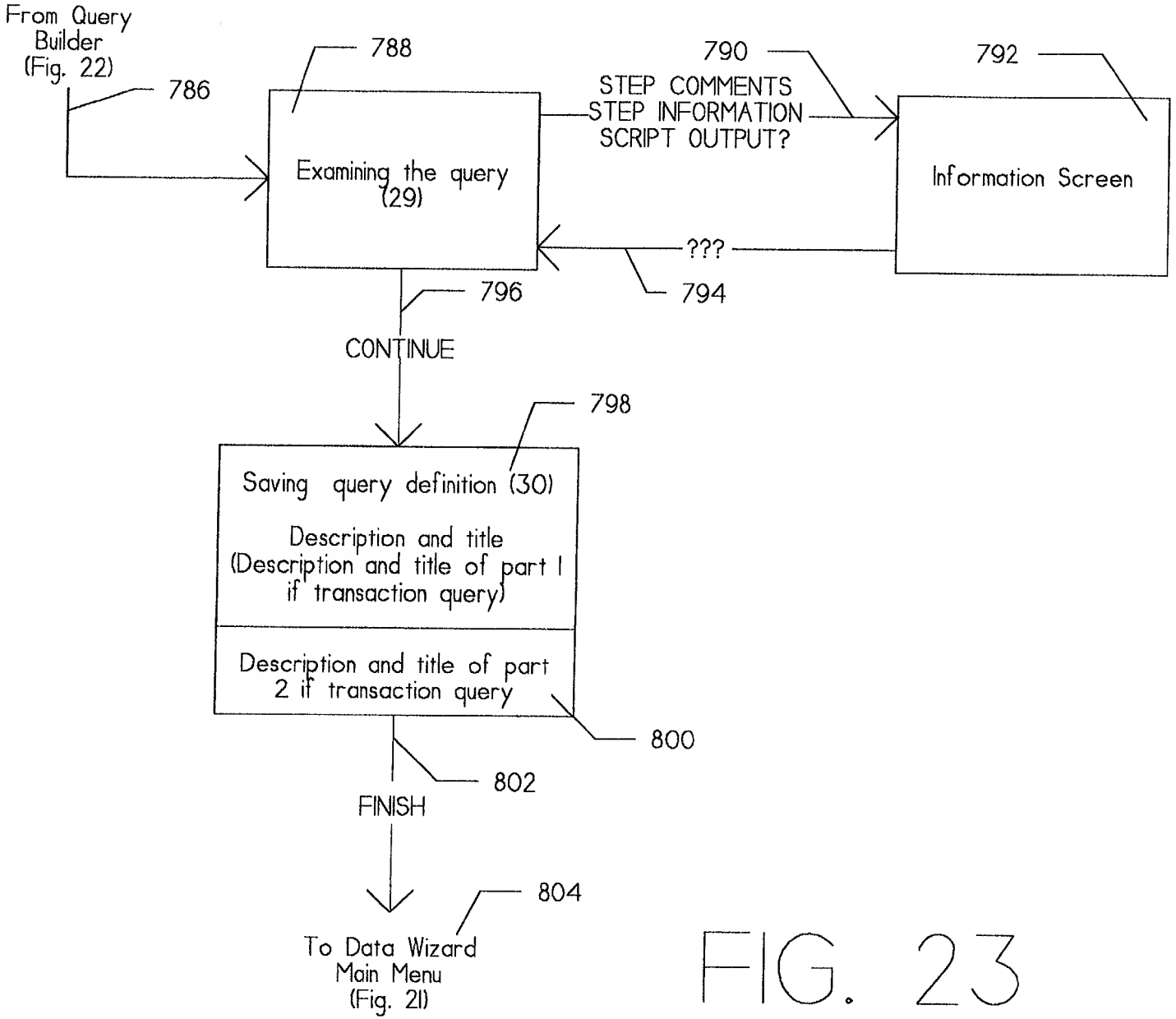
From Query Builder (Fig. 22)
786
788
Examining the query (29)
790
STEP COMMENTS
STEP INFORMATION
SCRIPT OUTPUT?
792
Information Screen
???
794
796
CONTINUE
798
Saving query definition (30)
Description and title (Description and title of part 1 if transaction query)
Description and title of part 2 if transaction query
800
802
FINISH
804
To Data Wizard Main Menu (Fig. 21)

FIG. 23

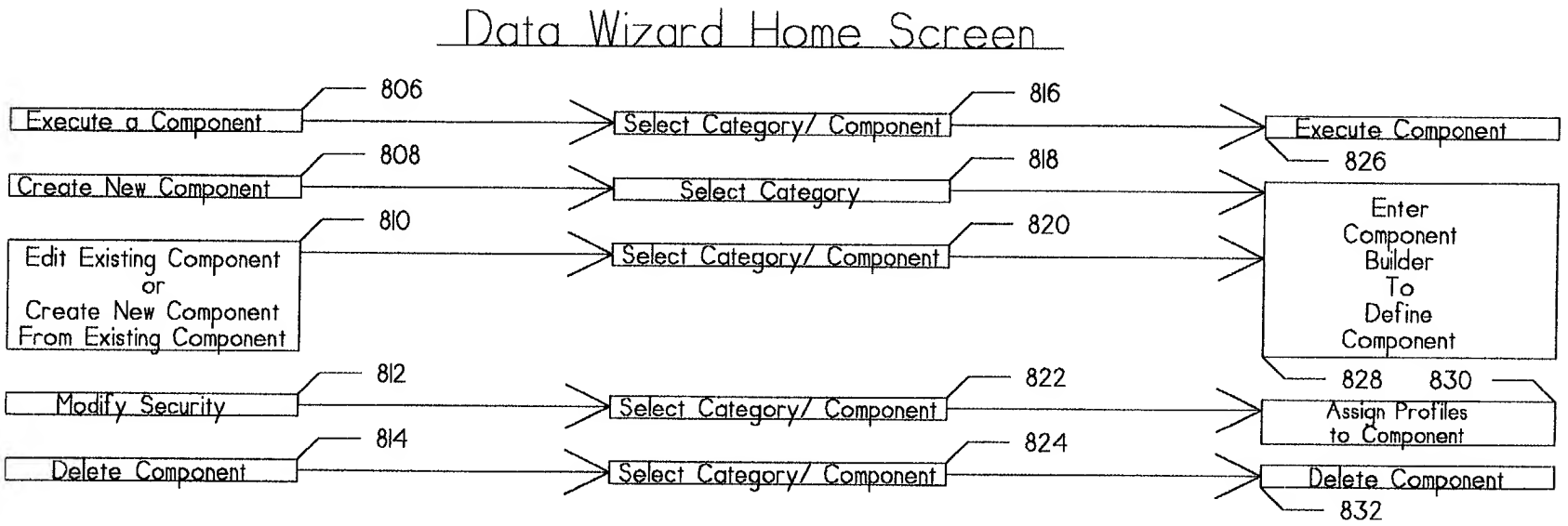
Data Wizard Home Screen
Execute a Component
806
Create New Component
808
Edit Existing Component
or
Create New Component
From Existing Component
810
Modify Security
812
Delete Component
814
Select Category/ Component
816
Select Category
818
Select Category/ Component
820
Select Category/ Component
822
Select Category/ Component
824
Execute Component
826
Enter
Component
Builder
To
Define
Component
828
830
Assign Profiles
to Component
Delete Component
832

FIG. 24

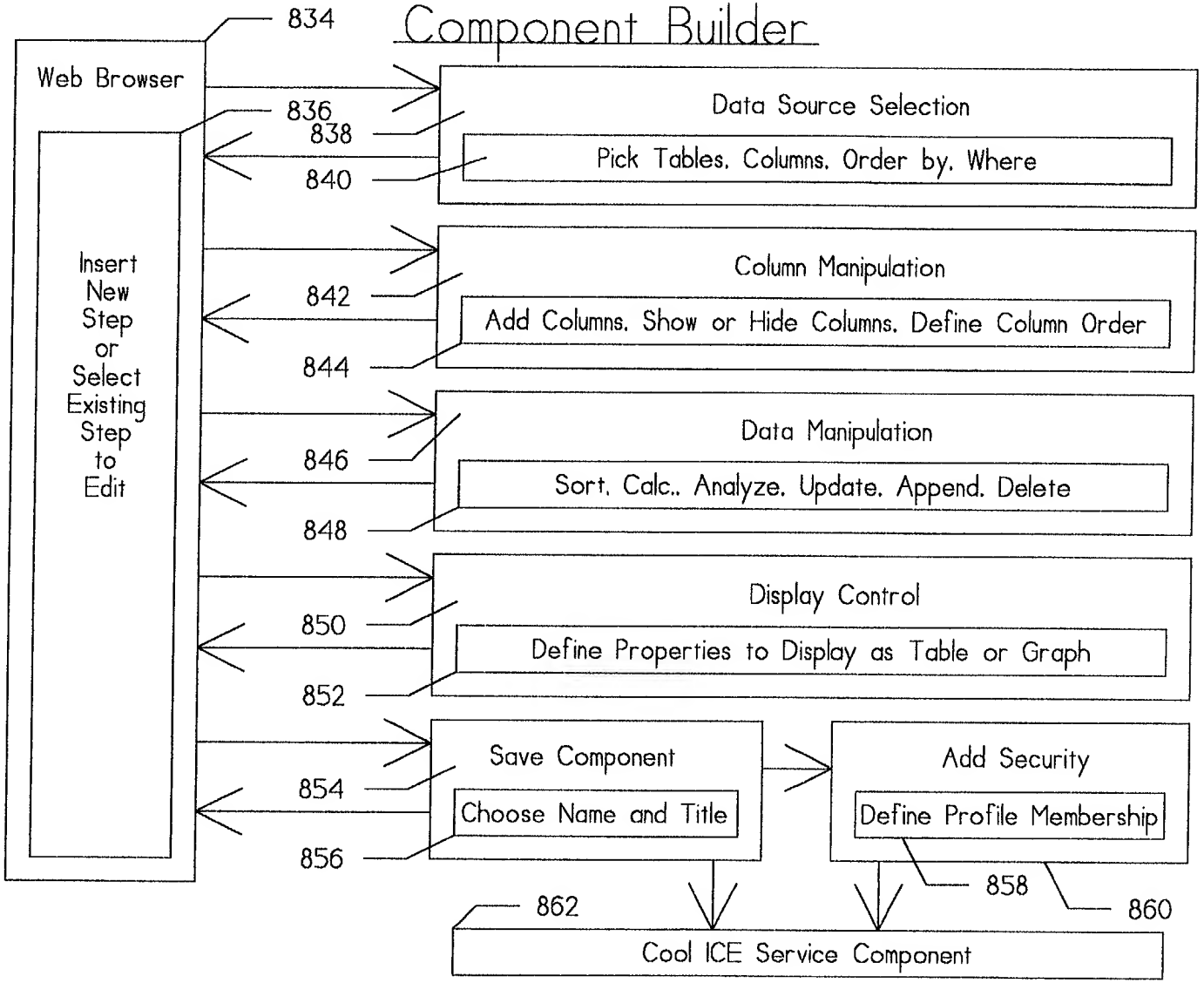
834
Component Builder
Web Browser
836
838
840
Data Source Selection
Pick Tables, Columns, Order by, Where
Insert New Step or Select Existing Step to Edit
842
844
Column Manipulation
Add Columns, Show or Hide Columns, Define Column Order
846
848
Data Manipulation
Sort, Calc., Analyze, Update, Append, Delete
850
852
Display Control
Define Properties to Display as Table or Graph
854
856
Save Component
Choose Name and Title
Add Security
Define Profile Membership
858
860
862
Cool ICE Service Component

FIG. 25

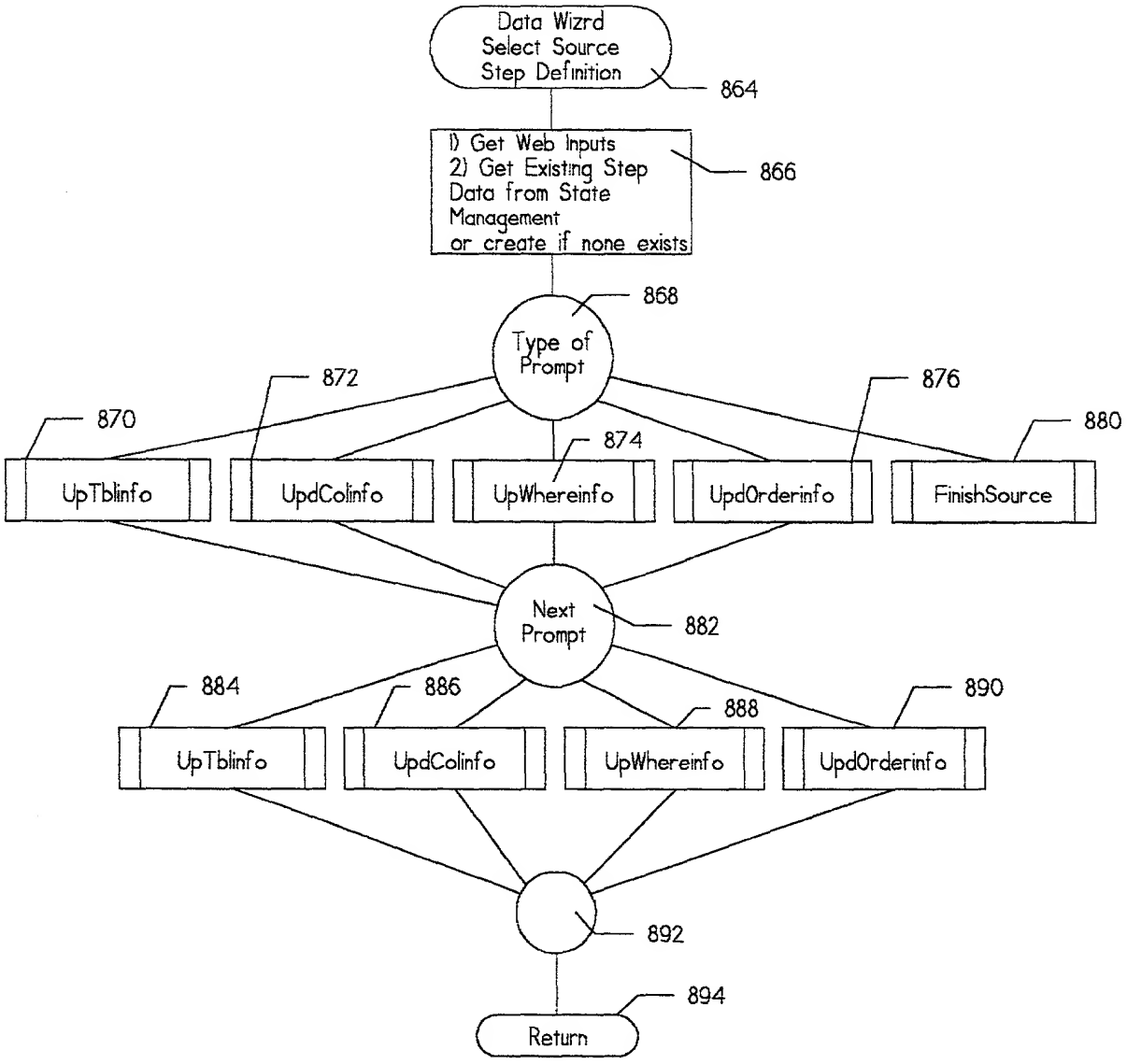
Data Wizrd
Select Source
Step Definition
864
1) Get Web Inputs
2) Get Existing Step
Data from State
Management
or create if none exists
866
868
Type of
Prompt
872
870
876
880
874
UpTblinfo
UpdColinfo
UpWhereinfo
UpdOrderinfo
FinishSource
Next
Prompt
882
884
886
888
890
UpTblinfo
UpdColinfo
UpWhereinfo
UpdOrderinfo
892
894
Return

FIG. 26

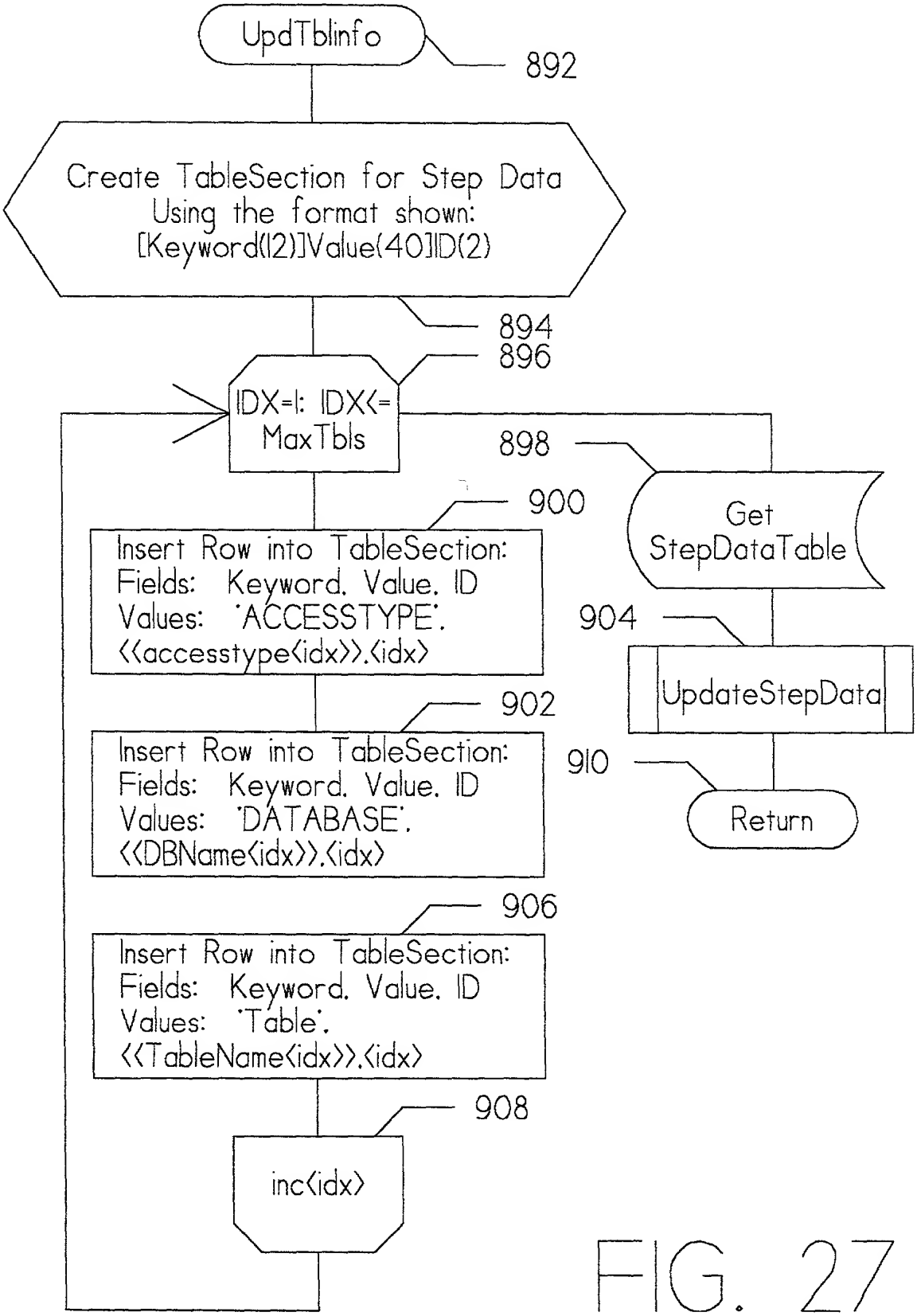
UpdTblinfo
892
Create TableSection for Step Data
Using the format shown:
[Keyword(12)]Value(40]ID(2)
894
896
IDX=1: IDX<=
MaxTbls
898
900
Insert Row into TableSection:
Fields: Keyword, Value, ID
Values: 'ACCESSTYPE',
<<accesstype<idx>>.<idx>
Get
StepDataTable
904
UpdateStepData
902
Insert Row into TableSection:
Fields: Keyword, Value, ID
Values: 'DATABASE',
<<DBName<idx>>.<idx>
910
Return
906
Insert Row into TableSection:
Fields: Keyword, Value, ID
Values: 'Table',
<<TableName<idx>>.<idx>
908
inc<idx>

FIG. 27

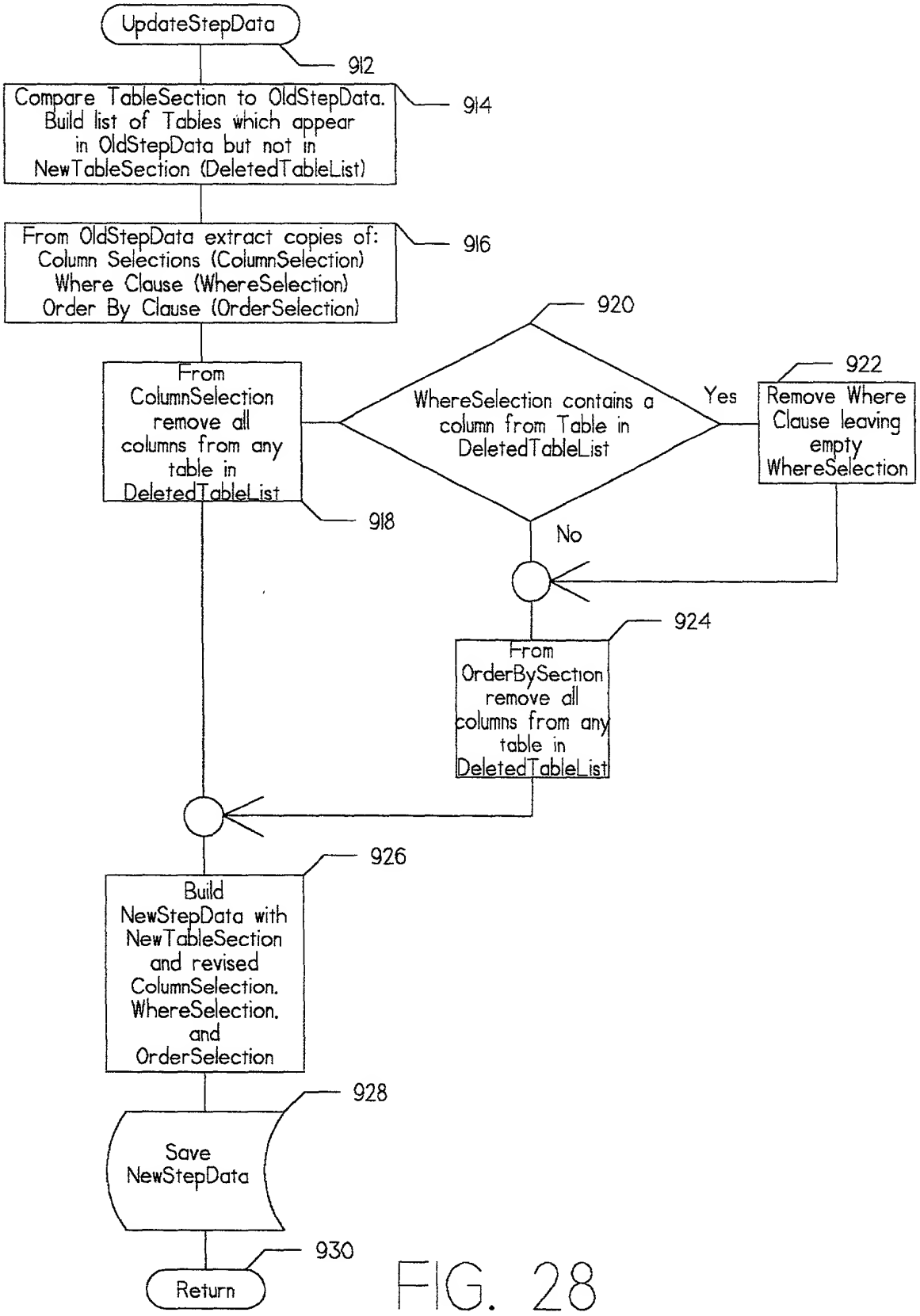
UpdateStepData
912
Compare TableSection to OldStepData. Build list of Tables which appear in OldStepData but not in NewTableSection (DeletedTableList)
914
From OldStepData extract copies of: Column Selections (ColumnSelection) Where Clause (WhereSelection) Order By Clause (OrderSelection)
916
920
WhereSelection contains a column from Table in DeletedTableList
Yes
922
Remove Where Clause leaving empty WhereSelection
From ColumnSelection remove all columns from any table in DeletedTableList
918
No
924
From OrderBySection remove all columns from any table in DeletedTableList
926
Build NewStepData with NewTableSection and revised ColumnSelection. WhereSelection. and OrderSelection
928
Save NewStepData
930
Return

FIG. 28

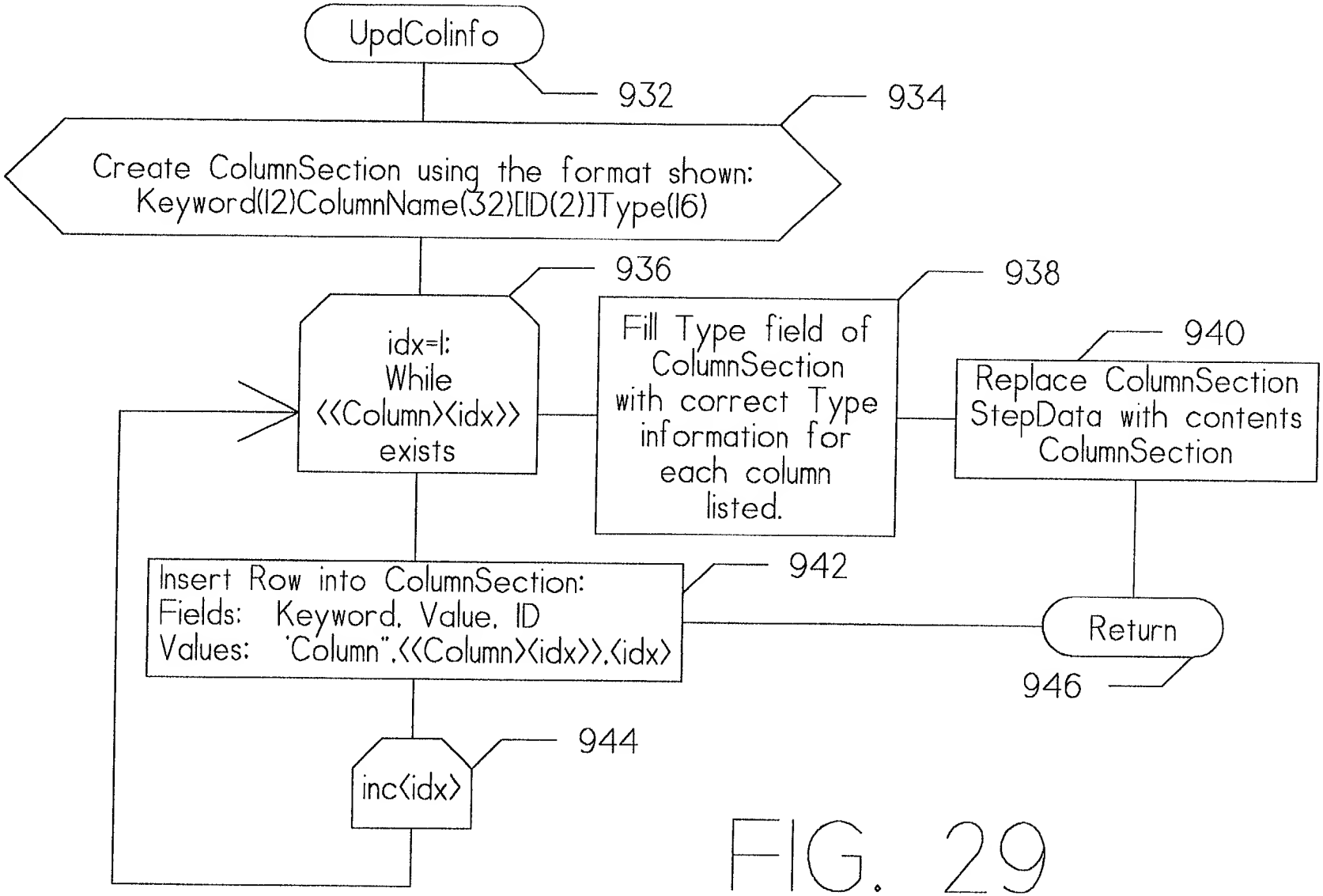
UpdColinfo
932
934
Create ColumnSection using the format shown:
Keyword(12)ColumnName(32)[ID(2)]Type(16)
936
idx=1:
While
<<Column><idx>>
exists
938
Fill Type field of
ColumnSection
with correct Type
information for
each column
listed.
940
Replace ColumnSection
StepData with contents
ColumnSection
Insert Row into ColumnSection:
Fields: Keyword, Value, ID
Values: 'Column',<<Column><idx>>,<idx>
942
Return
946
inc<idx>
944

FIG. 29

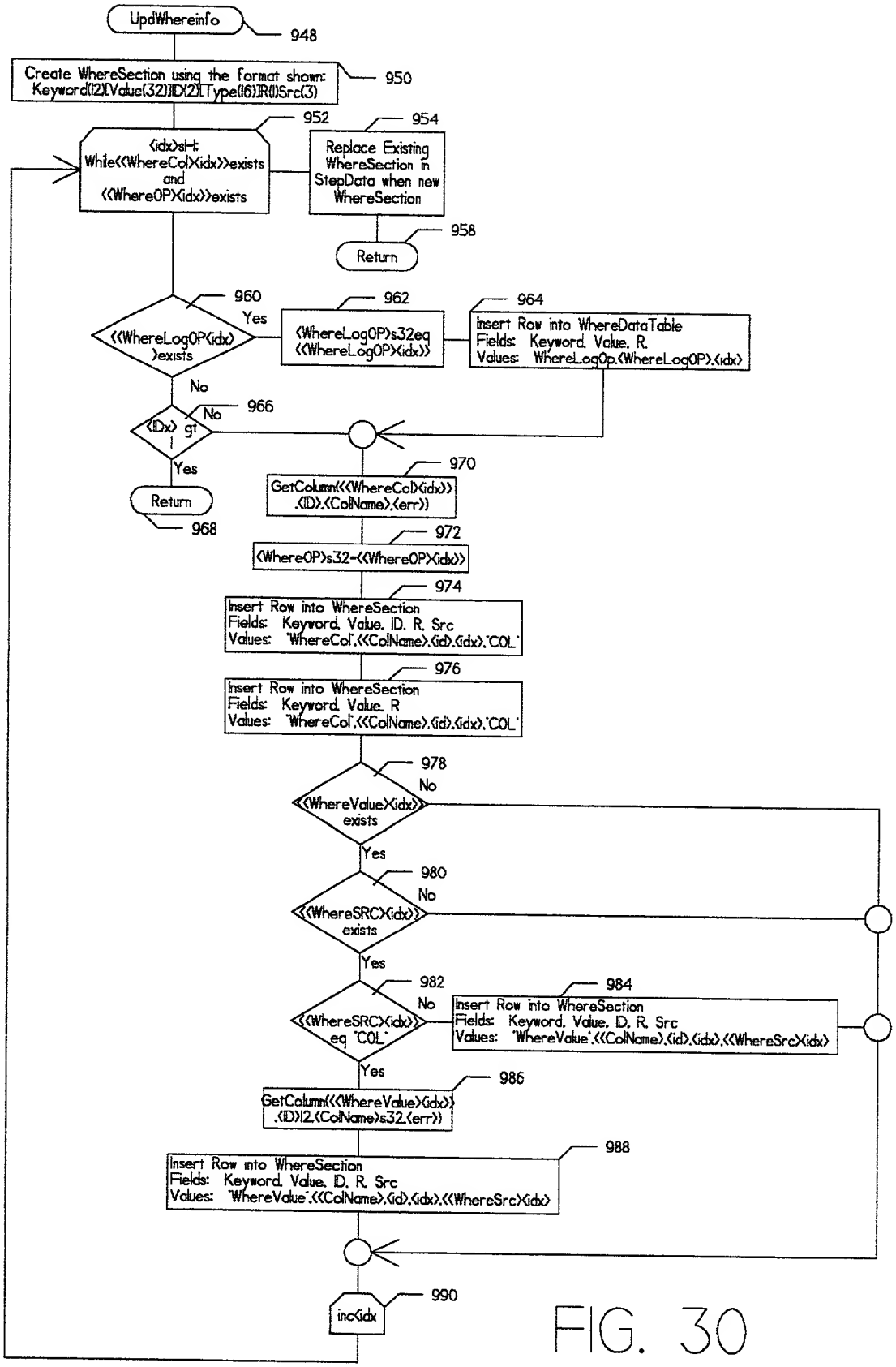
UpdWhereInfo
948
Create WhereSection using the format shown: Keyword(12)Value(32)ID(2)Type(16)R(1)Src(3)
950
952
954
<idx>si-1:
While(<<WhereCol>idx>>exists and <<WhereOP>idx>>exists
Replace Existing WhereSection in StepData when new WhereSection
958
Return
960
Yes
<<WhereLogOP>idx> >exists
962
<WhereLogOP>s32eq <<WhereLogOP>idx>>
964
Insert Row into WhereDataTable
Fields: Keyword, Value, R
Values: WhereLogOp,<WhereLogOP>,<idx>
No
<ID> gt 1
966
No
Yes
Return
968
970
GetColumn(<<WhereCol>idx>> ,<ID>,<ColName>,<err>)
972
<WhereOP>s32=<<WhereOP>idx>>
974
Insert Row into WhereSection
Fields: Keyword, Value, ID, R, Src
Values: 'WhereCol',<<ColName>,<id>,<idx>,'COL'
976
Insert Row into WhereSection
Fields: Keyword, Value, R
Values: 'WhereCol',<<ColName>,<id>,<idx>,'COL'
978
No
<<WhereValue>idx>> exists
Yes
980
No
<<WhereSRC>idx>> exists
Yes
982
No
<<WhereSRC>idx>> eq 'COL'
984
Insert Row into WhereSection
Fields: Keyword, Value, ID, R, Src
Values: 'WhereValue',<<ColName>,<id>,<idx>,<<WhereSrc>idx>
Yes
986
GetColumn(<<WhereValue>idx>> ,<ID>12,<ColName>s32,<err>)
988
Insert Row into WhereSection
Fields: Keyword, Value, ID, R, Src
Values: 'WhereValue',<<ColName>,<id>,<idx>,<<WhereSrc>idx>
990
incidx

FIG. 30

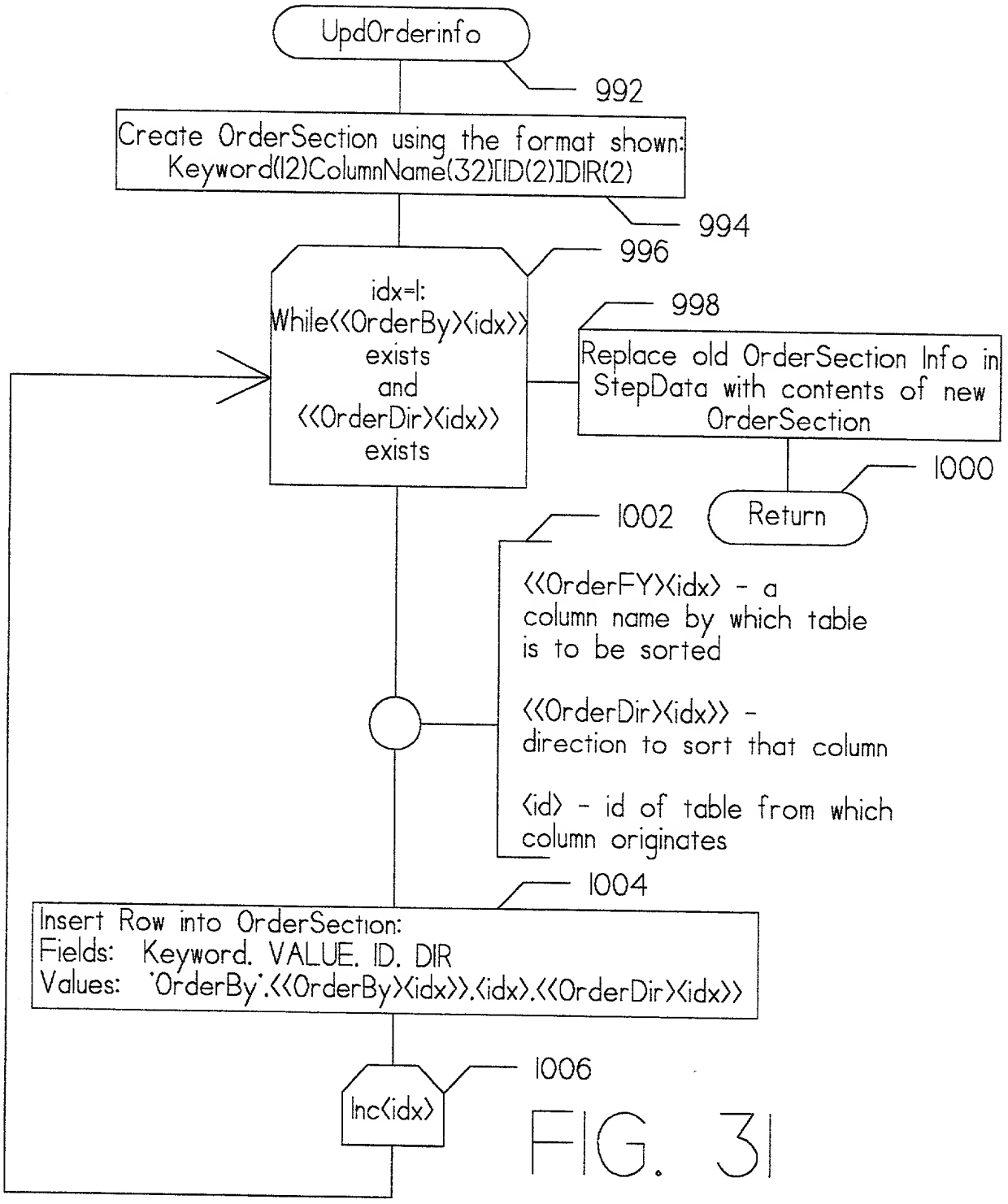
UpdOrderinfo
992
Create OrderSection using the format shown:
Keyword(12)ColumnName(32)[ID(2)]DIR(2)
994
996
idx=1:
While<<OrderBy>idx>>
exists
and
<<OrderDir>idx>>
exists
998
Replace old OrderSection Info in StepData with contents of new OrderSection
1000
Return
1002
<<OrderFY>idx> - a column name by which table is to be sorted
<<OrderDir>idx>> - direction to sort that column
<id> - id of table from which column originates
1004
Insert Row into OrderSection:
Fields: Keyword. VALUE. ID. DIR
Values: 'OrderBy'.<<OrderBy>idx>>.<idx>.<<OrderDir>idx>>
1006
Inc<idx>

FIG. 31

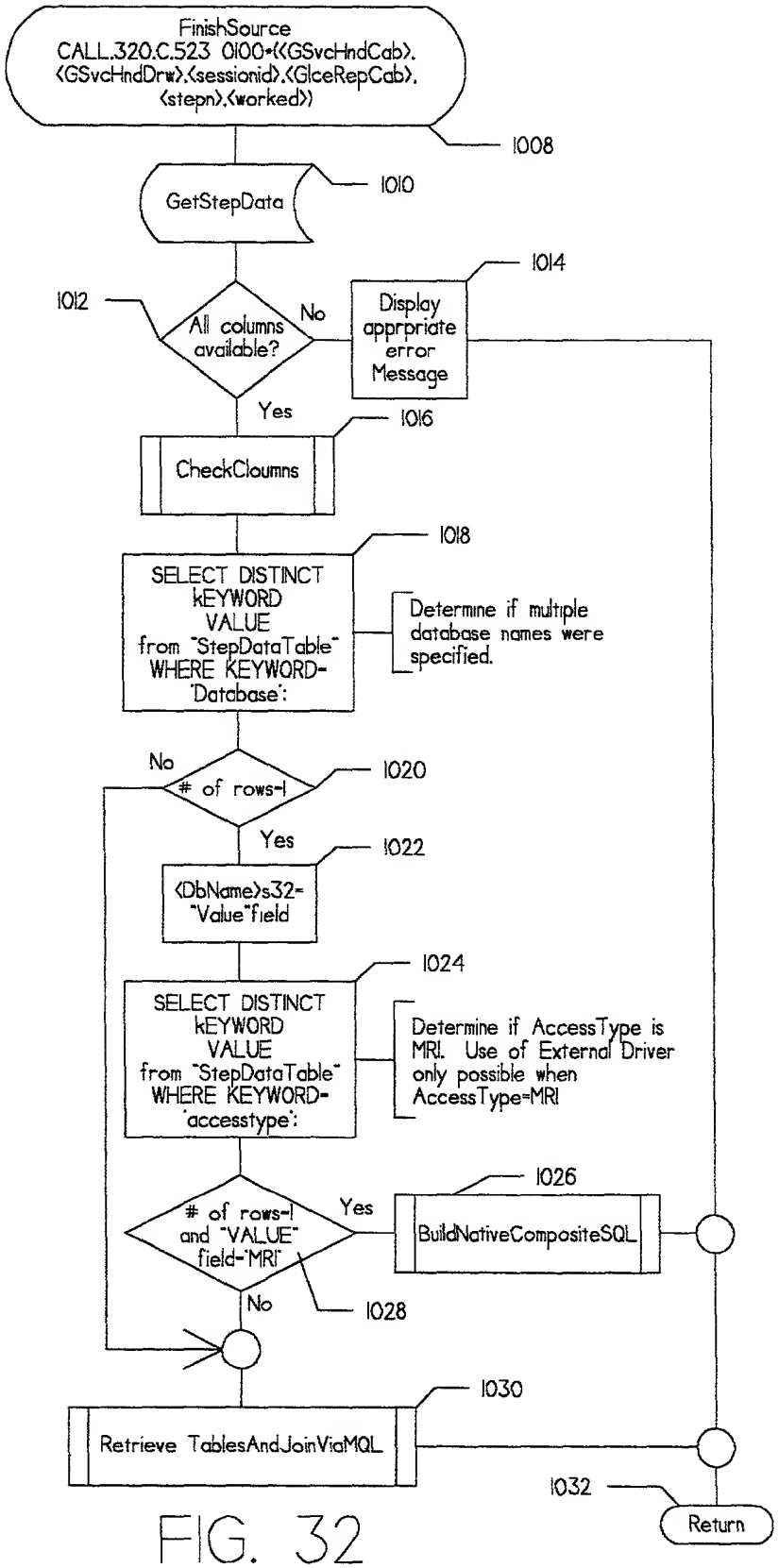
FinishSource
CALL.320.C.523 0100*(<GSvcHndCab>.
<GSvcHndDrw>.<sessionid>.<GlceRepCab>.
<stepn>.<worked>)
1008
GetStepData
1010
1012
All columns available?
No
1014
Display apprpriate error Message
Yes
1016
CheckCloumns
1018
SELECT DISTINCT kEYWORD VALUE from "StepDataTable" WHERE KEYWORD='Database';
Determine if multiple database names were specified.
No
# of rows=1
1020
Yes
1022
<DbName>s32="Value"field
1024
SELECT DISTINCT kEYWORD VALUE from "StepDataTable" WHERE KEYWORD='accesstype';
Determine if AccessType is MRI. Use of External Driver only possible when AccessType=MRI
# of rows=1 and "VALUE" field="MRI"
Yes
1026
BuildNativeCompositeSQL
No
1028
1030
Retrieve TablesAndJoinViaMQL
1032
Return

FIG. 32

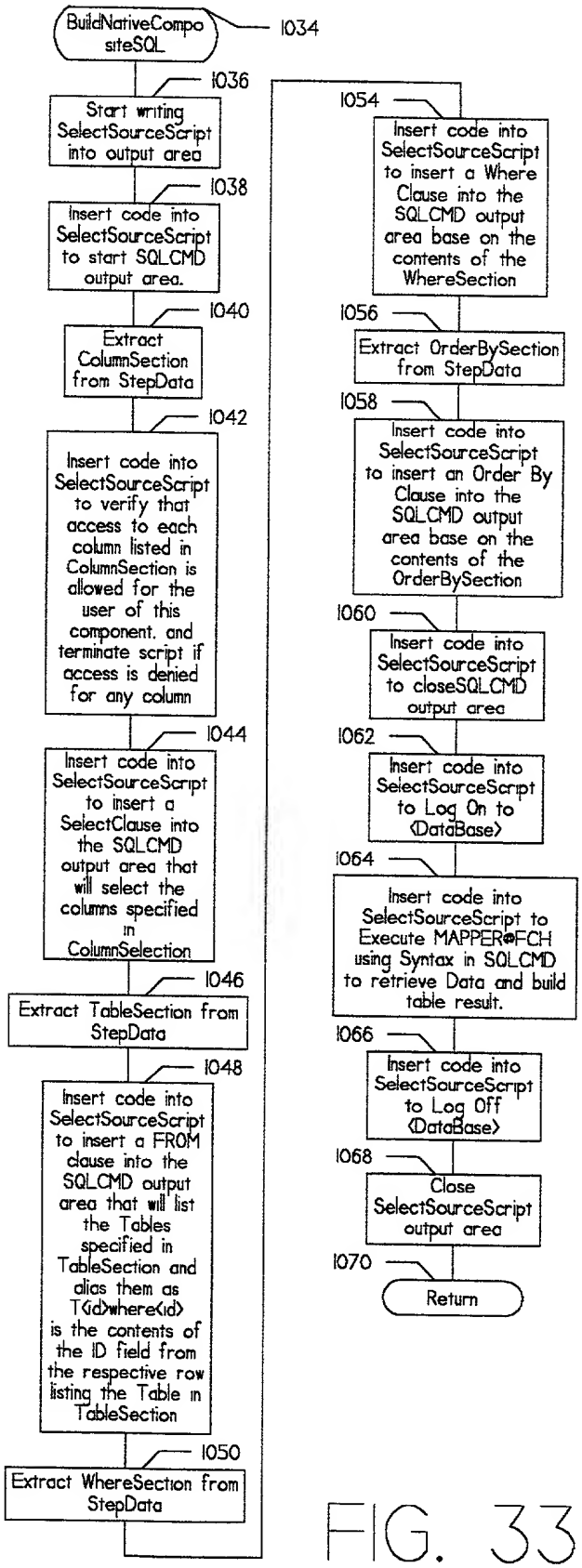
BuildNativeCompositeSQL
1034
1036
Start writing SelectSourceScript into output area
1038
Insert code into SelectSourceScript to start SQLCMD output area.
1040
Extract ColumnSection from StepData
1042
Insert code into SelectSourceScript to verify that access to each column listed in ColumnSection is allowed for the user of this component. and terminate script if access is denied for any column
1044
Insert code into SelectSourceScript to insert a SelectClause into the SQLCMD output area that will select the columns specified in ColumnSelection
1046
Extract TableSection from StepData
1048
Insert code into SelectSourceScript to insert a FROM clause into the SQLCMD output area that will list the Tables specified in TableSection and alias them as T<id>where<id> is the contents of the ID field from the respective row listing the Table in TableSection
1050
Extract WhereSection from StepData
1054
Insert code into SelectSourceScript to insert a Where Clause into the SQLCMD output area base on the contents of the WhereSection
1056
Extract OrderBySection from StepData
1058
Insert code into SelectSourceScript to insert an Order By Clause into the SQLCMD output area base on the contents of the OrderBySection
1060
Insert code into SelectSourceScript to closeSQLCMD output area
1062
Insert code into SelectSourceScript to Log On to <DataBase>
1064
Insert code into SelectSourceScript to Execute MAPPER@FCH using Syntax in SQLCMD to retrieve Data and build table result.
1066
Insert code into SelectSourceScript to Log Off <DataBase>
1068
Close SelectSourceScript output area
1070
Return

FIG. 33

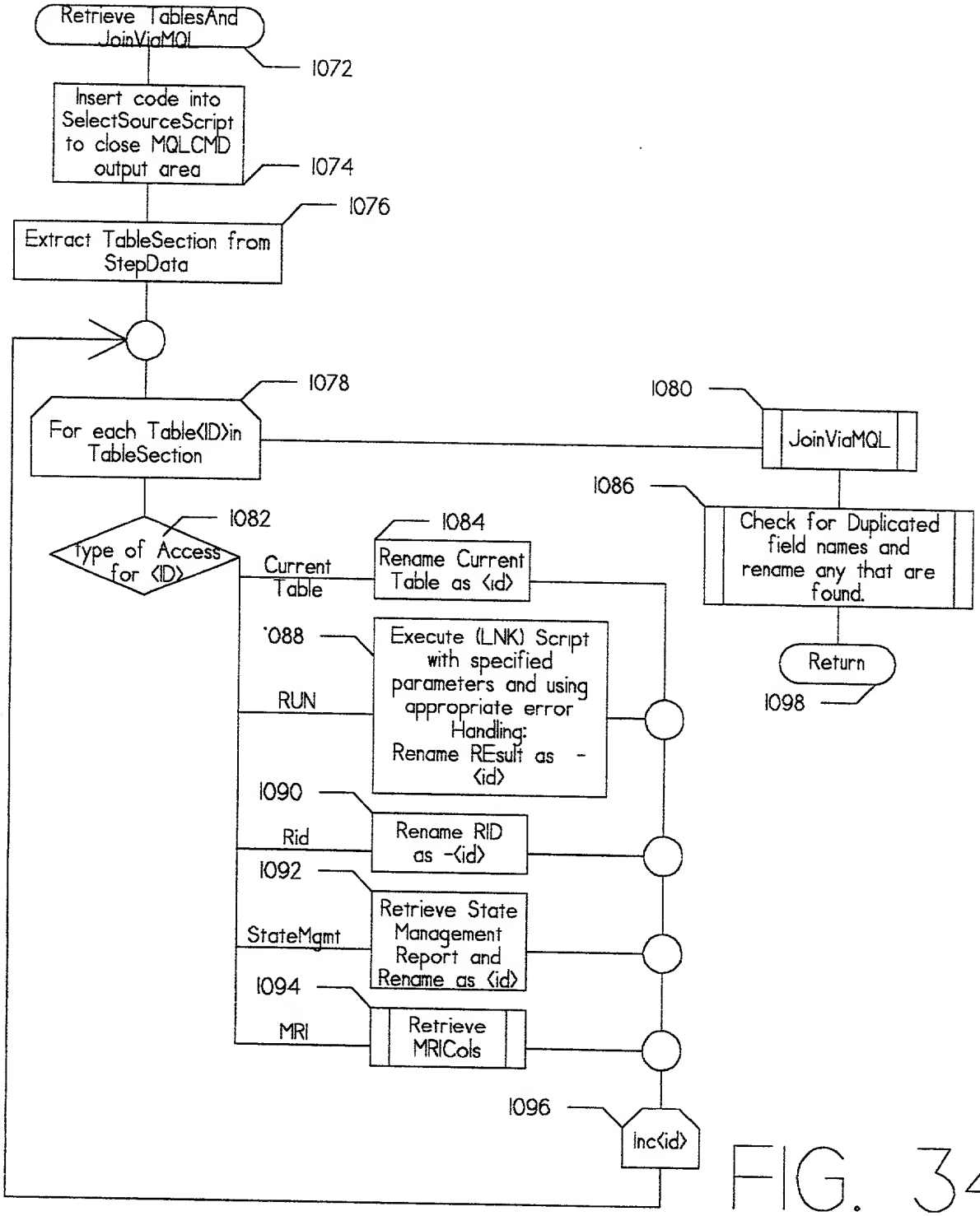
Retrieve TablesAnd JoinViaMQL
1072
Insert code into SelectSourceScript to close MQLCMD output area
1074
1076
Extract TableSection from StepData
1078
For each Table<ID>in TableSection
1080
JoinViaMQL
1086
Check for Duplicated field names and rename any that are found.
Return
1098
1082
Type of Access for <ID>
1084
Current Table
Rename Current Table as <id>
1088
RUN
Execute (LNK) Script with specified parameters and using appropriate error Handling: Rename REsult as - <id>
1090
Rid
Rename RID as -<id>
1092
StateMgmt
Retrieve State Management Report and Rename as <id>
1094
MRI
Retrieve MRICols
1096
Inc<id>

FIG. 34

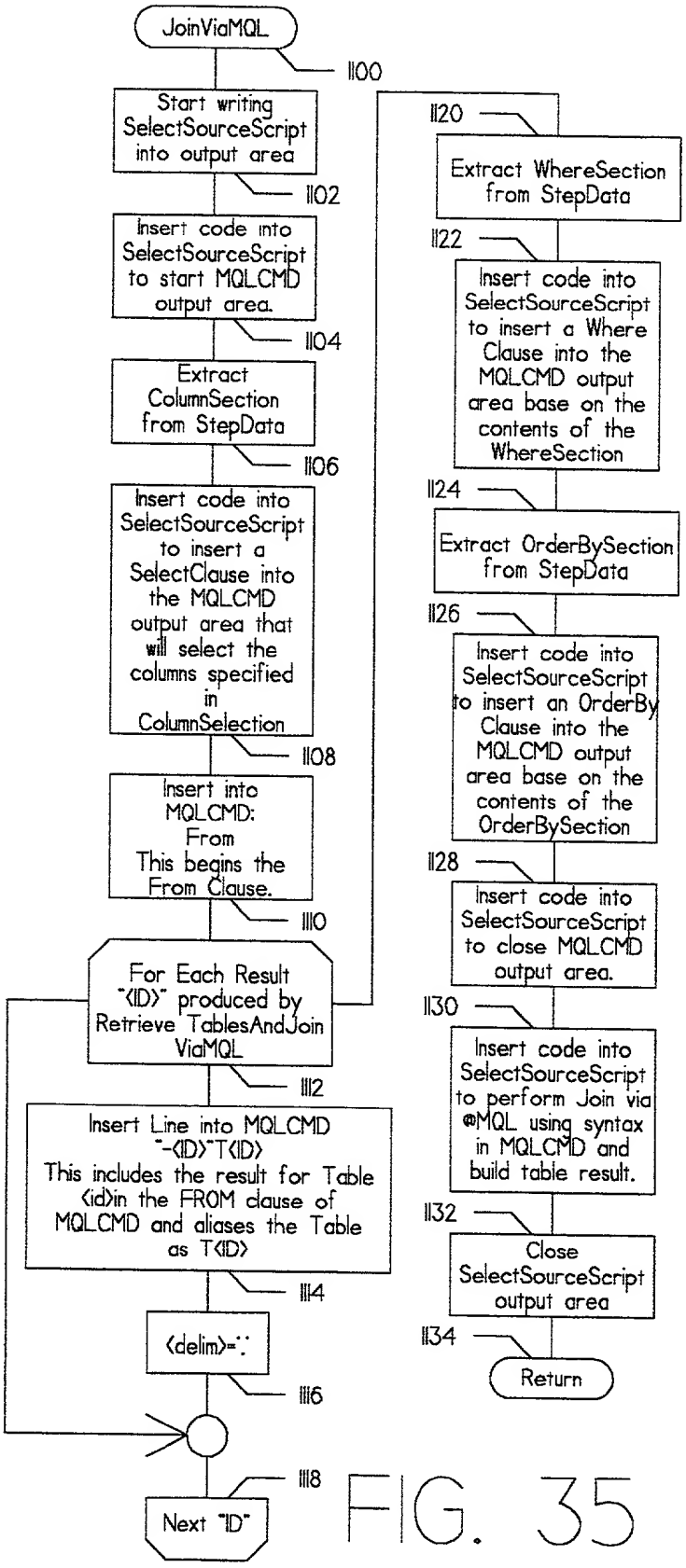
JoinViaMQL
1100
Start writing SelectSourceScript into output area
1102
Insert code into SelectSourceScript to start MQLCMD output area.
1104
Extract ColumnSection from StepData
1106
Insert code into SelectSourceScript to insert a SelectClause into the MQLCMD output area that will select the columns specified in ColumnSelection
1108
Insert into MQLCMD: From This begins the From Clause.
1110
For Each Result "<ID>" produced by Retrieve TablesAndJoin ViaMQL
1112
Insert Line into MQLCMD "-<ID>"T<ID> This includes the result for Table <id>in the FROM clause of MQLCMD and aliases the Table as T<ID>
1114
<delim>=":"
1116
1118
Next "ID"
1120
Extract WhereSection from StepData
1122
Insert code into SelectSourceScript to insert a Where Clause into the MQLCMD output area base on the contents of the WhereSection
1124
Extract OrderBySection from StepData
1126
Insert code into SelectSourceScript to insert an OrderBy Clause into the MQLCMD output area base on the contents of the OrderBySection
1128
Insert code into SelectSourceScript to close MQLCMD output area.
1130
Insert code into SelectSourceScript to perform Join via @MQL using syntax in MQLCMD and build table result.
1132
Close SelectSourceScript output area
1134
Return

FIG. 35